arXiv:2604.22240v2 [cs.CV] 19 Aug 2026

OccDirector: Language-Guided Behavior and Interaction Generation in 4D Occupancy Space

Zhuding Liang[1⋆], Tianyi Yan[1*], Dubing Chen[1], Jiasen Zheng[2], Huan Zheng[1], Cheng-zhong Xu[1], Yida Wang[3], Kun Zhan[3], and Jianbing Shen[1,✉]

[1] SKL-IOTSC, CIS, University of Macau, Macau, China
[2] Northwestern University, Evanston, United States
[3] Li Auto Inc, Beijing, China

Abstract. Generative world models increasingly rely on 4D occupancy for realistic autonomous driving simulation. However, existing generation frameworks are limited in two respects: geometric-condition-based methods depend on explicit trajectories and cannot generalize to free-form instructions, while text-based methods operate at a coarse attribute level and fail to orchestrate complex, sequential multi-agent interactions. We propose OccDirector, a framework that generates 4D occupancy dynamics conditioned solely on natural language scripts, without requiring any geometric priors. OccDirector encodes free-form scripts via a frozen VLM and processes occupancy tokens through a Spatio-Temporal MMDiT backbone. A history-prefix anchoring strategy further ensures that generated sequences remain consistent with any provided context over long horizons. To support training and evaluation, we introduce OccInteract-85k, a dataset of 85k clips annotated with multi-level language instructions spanning static layouts to intricate multi-agent behaviors, alongside a novel VLM-based evaluation benchmark. Extensive experiments demonstrate state-of-the-art generation quality and strong instruction-following fidelity across diverse driving scenarios.

Keywords: 4D Occupancy Generation · Language-Guided Scene Generation · Multi-Agent Interaction Modeling · Vision-Language Model Conditioning · Spatio-Temporal Transformer

1 Introduction

The evolution of autonomous driving (AD) systems [6,9,23] is increasingly relying on Generative World Models [11,34,58] that can simulate diverse and realistic driving scenarios for closed-loop training and validation. Among various scene

⋆ Equal Contribution.

✉ Corresponding author: *Jianbing Shen*. This work was supported by the Science and Technology Development Fund of Macau SAR (FDCT) under grants 0134/2025/RIA2 and 0102/2023/RIA2, and the Jiangyin Hi-tech Industrial Development Zone under the Taihu Innovation Scheme (EF2025-00003-SKL-IOTSC).

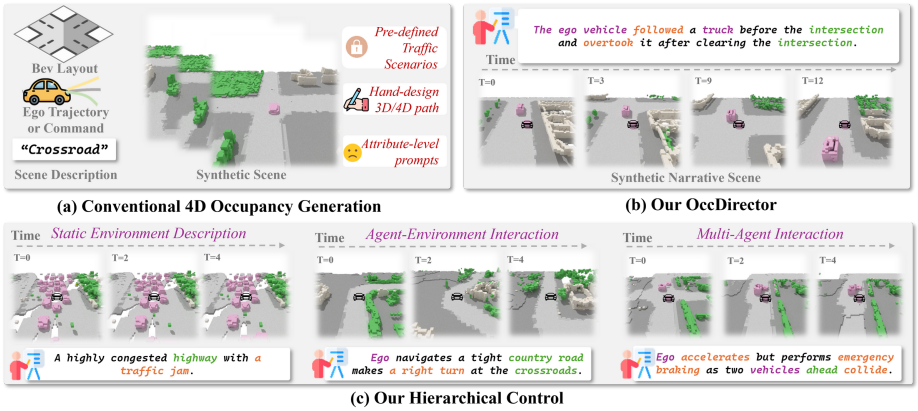


Fig. 1: Script-driven behavior and interaction generation with OccDirector. (a) vs. (b): Comparison between previous methods and our script-driven approach. (c) Hierarchical generation capabilities across Static, Agent-Environment, and Multi-Agent levels. The ego vehicle is marked with a *black car icon. Note: Due to raw data limitations Sec. 4.1, some results may exhibit missing ego voxels or sensor blind spots.*

representations, 4D Occupancy (space-time occupancy) [17, 28] has emerged as the most unified and comprehensive representation. Unlike 3D bounding boxes which oversimplify object geometry [39, 41], or LiDAR point clouds [1, 46] which suffer from sparsity and occlusion, 4D occupancy grids offer a dense, geometry-aware, and temporally consistent description of dynamic environments. Consequently, synthesizing high-fidelity 4D occupancy rollouts has become a critical frontier in the computer vision community [8, 55].

Despite impressive progress in 3D/4D generation, existing occupancy generation frameworks [3, 22, 44] heavily rely on explicit geometric conditions—such as BEV semantic layouts, discrete navigation command, or predefined ego-vehicle trajectories—to control the generative process. As illustrated in Fig. 1(a), methods like [3, 14, 36] depend on rigid explicit trajectories or simplistic attribute prompts. While offering precise spatial constraints, these conditions are fundamentally inflexible and unscalable. Users must laboriously hand-design accurate 3D paths, a process that cannot intuitively express high-level intents. In contrast, our OccDirector (Fig. 1(b,c)) acts as a "scenario director", interpreting complex natural language scripts to orchestrate dynamic driving scenes, such as *"congested traffic jam"* or *"emergency braking"*. To date, generating dynamic 4D occupancy purely guided by natural language remains an unsolved challenge.

Pioneering text-driven 4D occupancy generation faces a significant challenge: the Semantic-Spatiotemporal Gap. Language instructions inherently rely on relational structures and sequential logic to describe abstract behaviors. For instance, consider the instruction: *"The ego vehicle followed a truck before the intersection and overtook it after clearing the intersection."* Translating such high-

level temporal logic into occupancy generation is non-trivial, as it demands rigorous spatio-temporal consistency and physically plausible evolution at the voxel level. However, most existing autonomous driving diffusion models [12, 45, 58] fail to bridge this gap. Relying on generic text encoders [31, 32] that function largely as "bag-of-words" extractors, these models lack the compositional reasoning required to parse complex interactions [19, 56]. Consequently, they often produce implausible collisions, discontinuous motions, or topological violations when processing intricate instructions [24].

To this end, we introduce OccDirector, taking a decisive step toward language-driven scenario orchestration. Unlike prior works constrained by geometric priors, OccDirector establishes a new paradigm of purely text-guided 4D occupancy generation, bridging the gap from appearance synthesis to complex agent behavior and interaction generation. By replacing rigid trajectory inputs with a director-like natural language interface, OccDirector enables comprehensive controllability along three complementary axes, as visualized in Fig. 1(c): (i) **Static Environment**, synthesizing diverse topologies like *"traffic jams"* via semantic composition; (ii) **Agent-Environment Interaction**, executing topology-constrained maneuvers such as yielding at intersections; and (iii) **Multi-Agent Interaction**, generating physically plausible reactive behaviors like *"emergency braking to avoid collisions"*. Technically, OccDirector leverages a powerful Vision-Language Model (VLM) as the text encoder to deeply comprehend procedural semantics. The joint text-occupancy sequence is modeled using a scalable spatio-temporal MMDiT equipped with Spatio-Temporal Separated Attention (STSA) and a history-prefix token-replace strategy for interaction-consistent rollouts. Furthermore, to overcome the complete absence of paired text-4D occupancy data, we develop an automated data engine that translates real-world logs and simulator data into abundant, logic-rich occupancy-language pairs.

Our contributions can be summarized as follows:

- We propose OccDirector, a pioneering framework for text-guided 4D occupancy generation, enabling director-like control over agent behaviors without rigid geometric priors.
- We design a VLM-driven Spatio-Temporal MMDiT with STSA and a history-prefix anchoring strategy, effectively bridging the semantic-spatiotemporal gap for consistent interactions.
- We introduce OccInteract-85k, a large-scale dataset with hierarchical language instructions ranging from static layouts to complex interactions, alongside a novel VLM-based benchmark.
- Extensive experiments demonstrate that OccDirector achieves state-of-the-art generation quality and unprecedented instruction-following capabilities.

2 Related Work

2.1 Generative World Models for Autonomous Driving

Generative world models have emerged as a paradigm shift in autonomous driving simulation, enabling the synthesis of diverse driving scenarios for closed-loop

training [16, 34, 45, 58]. Early approaches predominantly focused on 2D Video Generation [12,13,18], leveraging diffusion models to produce photorealistic camera views. While visually compelling, these perspective-view methods often lack explicit 3D geometric consistency and struggle to support downstream planning tasks that require precise spatial reasoning.

Recently, 4D Occupancy has been established as a unified representation bridging visual fidelity and geometric explicitness [41, 51]. By representing the world as a dense, voxelized spatiotemporal volume, it supports physically grounded simulation. However, existing occupancy world models [57, 59] primarily focus on *predicting* future states from past observations (forecasting). They lack the generative flexibility to create new scenarios from scratch based on user intent, limiting their utility for controllable simulation.

2.2 Conditioned 4D Occupancy Generation

Unlike forecasting, occupancy generation aims to synthesize plausible 4D scenes conditioned on specific inputs. Current state-of-the-art frameworks heavily rely on Explicit Geometric Conditions. For instance, methods like OccSora [44] and DynamicCity [3] condition the diffusion process on HD maps, 3D semantic layouts, or pre-defined agent trajectories. While these strong priors ensure structural validity, they impose a rigid generation paradigm: users must provide precise, low-level geometric inputs (e.g., drawing a spline for a lane change) to control the output. This "trajectory-engineering" process is labor-intensive and non-intuitive. In contrast, OccDirector proposes a paradigm shift from geometry-driven to language-driven generation. We eliminate the need for pre-defined trajectories, allowing users to orchestrate scene evolution through high-level semantic scripts (e.g., *"an aggressive cut-in"*), thereby offering a more scalable interface for scenario design.

2.3 Text-Conditioned Scene Generation and Fine-Grained Control

The integration of natural language into generative models has significantly advanced image and video synthesis [26, 35, 40, 43]. In the autonomous driving domain, however, language control remains largely limited to high-level attributes. Existing works [7, 11, 30, 58] typically utilize text prompts to control Global Attributes (e.g., weather conditions, time-of-day) or static scene composition. While some approaches like UniScene [22] incorporate text, they primarily use it for texture rendering or style transfer, leaving the core spatiotemporal dynamics (i.e., how agents move) determined by other modalities. Other works like X-Scene [54] adopt a hierarchical approach, converting text into intermediate layouts or bounding boxes before generation. Although this bridges language and static layout, it may struggle to capture the continuous nuances of dynamic interactions.

In this work, we introduce OccDirector, which pioneers end-to-end text-to-4D occupancy generation. Unlike methods that treat text as a style modifier or rely on intermediate layout proxies, we learn a direct mapping from procedural

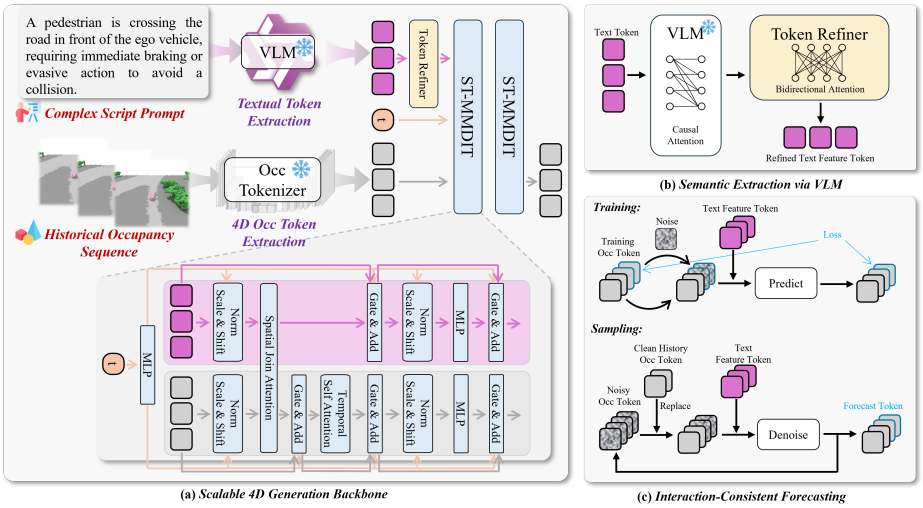


Fig. 2: Model Architecture of OccDirector. (a) Scalable 4D Generation Backbone: A Spatio-Temporal MMDiT processes text and occupancy tokens via dual streams, utilizing separated spatial and temporal attention. **(b) Semantic Extraction:** A frozen VLM and a Token Refiner bridge the semantic-spatial gap. **(c) Interaction-Consistent Forecasting:** The history-anchoring strategy ensures the generated future strictly adheres to the context during both training and sampling.

language instructions to continuous spatiotemporal voxel flows, enabling fine-grained control over multi-agent interactions and maneuvers purely from natural language.

3 OccDirector: Language-Conditioned 4D Occupancy Interaction Generation

3.1 Method Overview

OccDirector is designed for *script-driven behavior and interaction generation* within 4D occupancy rollouts (Sec. 1). Formally, given a natural language script π and an optional history context $\mathbf{O}_{1:h}$, our goal is to generate a future 4D occupancy sequence $\mathbf{O}_{h+1:F}$ that strictly adheres to the spatial layout and dynamic interaction intents specified in π. Fig. 2 illustrates our architecture.

3.2 Joint Latent Representation

Directly modeling raw 4D occupancy data and complex natural language is computationally intractable and semantically challenging. To map both modalities into a unified, efficient latent space, we formulate our input encoding stage with three specialized components.

Geometry-Aware Occupancy Tokenization. To alleviate the extreme memory and computational footprint of 4D occupancy sequences $\mathbf{O}_{1:F} \in \{0, \dots, K-1\}^{F \times H \times W \times D}$, we require a geometry-aware compression mechanism. Building upon the discrete auto-encoding principles of DOME [14], we adopt a frozen Occ-VAE. This module maps the discrete 4D rollout into a compact, continuous latent grid, which is subsequently patchified and projected into a sequence of tokens $\mathbf{H} \in \mathbb{R}^{F \times S \times d_{\text{model}}}$, where F is the temporal length and S denotes the number of 3D spatial tokens per frame. Unlike DOME's implementation, we pretrain the Occ-VAE with variable-length rollouts, enabling flexible generation horizons.

Procedural Semantic Extraction via VLM. Effective script-driven generation requires controlling dynamic *interaction intents* (e.g., yielding, cut-in) over time. Standard CLIP-style embeddings [31,32] often lack the relational and procedural semantics needed for such fine-grained control. Recent advancements have demonstrated that leveraging LLMs or VLMs for text feature embedding yields superior performance compared to CLIP-style embeddings [40,49,50,53]. To capture these complex global relations, as shown in Fig. 2(b), we utilize a frozen Vision-Language Model (VLM) text encoder (Qwen3-VL-2B [52]). Given a prompt π, the encoder extracts dense, high-dimensional representations $\tilde{\mathbf{C}} \in \mathbb{R}^{L \times d_{\text{text}}}$, where L is the sequence length.

Token Refiner. While VLM features provide rich semantics, they are inherently suboptimal for direct generative control in 4D spatial-temporal layouts due to modality gaps. To bridge this semantic-spatial divide, we introduce a lightweight bidirectional token refiner (Fig. 2(b)) [20, 27]. This module, denoted as Refine$(\cdot)$, employs a shallow transformer alternating between full self-attention and MLP blocks. It projects the raw features to the model dimension, $\mathbf{C} = \text{Refine}(\tilde{\mathbf{C}}) \in \mathbb{R}^{L \times d_{\text{model}}}$, producing text tokens with enhanced compositional alignment for occupancy generation.

3.3 Scalable 4D Generation Backbone

Modeling environment constraints and multi-agent interactions in 4D space demands capturing long-range dependencies; failure to do so results in accumulated errors, such as geometry violations or implausible discontinuities. However, the cubic complexity of 3D spatial tokens (S) multiplied by the temporal dimension (F) makes standard full attention computationally prohibitive.

To resolve this scalability bottleneck while maintaining strict text-to-layout alignment, we formulate a double-stream transformer backbone (Fig. 2(a)) that builds upon the MMDiT architecture [10]. This design preserves a dedicated text pathway, allowing for repeated, deep interactions between text tokens $\mathbf{C}$ and occupancy tokens $\mathbf{H}$ to strengthen fine-grained conditioning.

To efficiently process the massive number of occupancy tokens, we factorize the attention mechanism via Spatio-Temporal Separated Attention (STSA) [29]. Specifically, each transformer block decomposes the operation into two sequential steps: (i) *Frame-wise 3D spatial joint attention*, which models interactions between text tokens and spatial occupancy tokens within each independent frame

to ensure script-to-layout grounding; followed by (ii) *Patch-wise temporal self-attention*, applied solely across the temporal axis of occupancy tokens to propagate dynamics smoothly over time. To align with this factorization, we apply 2D spatial RoPE [37] on the $H \times W$ grid within each frame (with the Z-axis folded into the channel dimension), and 1D temporal RoPE across frames.

3.4 Interaction-Consistent Forecasting via History Anchoring

To train OccDirector for stable and high-quality continuous latent generation, we adopt the Continuous Normalizing Flow (CNF) framework via Flow Matching [25]. We define $t = 0$ as the ground-truth clean data and $t = 1$ as the pure noise. Let $\mathbf{x}_0$ denote the ground-truth latent occupancy tokens and $\mathbf{x}_1 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ be the standard Gaussian noise. We construct a linear probability path $\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1$, where $t \sim \mathcal{U}(0,1)$. The corresponding vector field is given by $v_t(\mathbf{x}) = \mathbf{x}_1 - \mathbf{x}_0$. The base objective trains the model v_θ to regress this vector field conditioned on the refined text tokens $\mathbf{C}$.

Training with History-Prefix Anchoring. For behavior generation, users often specify partial layouts or history contexts $\mathbf{O}_{1:h}$. Maintaining strict consistency with these constraints is critical, as small deviations in early frames can compound into collisions or interaction drift. Rather than relying on training-free heuristics during inference (which often yield temporal inconsistencies), we explicitly equip the model with forecasting capabilities during the training phase.

To achieve this, we introduce a stochastic *token-replace anchoring* strategy, visualized in Fig. 2(c) [20]. During training, we randomly sample a history horizon h. With a certain probability (similar to classifier-free guidance dropping), we replace the noisy history prefix $\mathbf{x}_t^{(1:h)}$ with the perfectly clean ground-truth tokens $\mathbf{x}_0^{(1:h)}$. Let $\tilde{\mathbf{x}}_t$ denote this partially clean, partially noisy input sequence. To ensure the model focuses entirely on forecasting the unknown dynamics, we apply a masked loss that only penalizes the predictions on the future frames $(h+1:F)$:

$$\mathcal{L}_{\text{Anchor}} = \mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_1}\left[\left\| v_\theta(\tilde{\mathbf{x}}_t, \mathbf{C}, t)^{(h+1:F)} - (\mathbf{x}_1 - \mathbf{x}_0)^{(h+1:F)} \right\|_2^2\right]. \tag{1}$$

This formulation forces the network to learn the conditional transition dynamics from a clean history to a noisy future, strictly guided by the script $\mathbf{C}$.

Inference via Step-wise Anchoring. During the inference (ODE sampling) phase, we start from pure noise $\mathbf{x}_1$ and integrate backwards to $t = 0$. To enforce robust controllability and ensure the generated sequence perfectly matches the user-provided history $\mathbf{H}_{1:h}$, we apply step-wise anchoring. At every integration step t, after the solver updates the sequence, we forcefully overwrite the history prefix with the clean condition $\mathbf{x}_{t-\Delta t}^{(1:h)} \leftarrow \mathbf{H}_{1:h}$. Because the model has been explicitly trained to condition on clean history prefixes via Eq. (1), this step-wise replacement seamlessly anchors the generation process without introducing out-of-distribution artifacts, ensuring that the future dynamics $\mathbf{x}_0^{(h+1:F)}$ remain strictly coherent with the provided context.

4 OccInteract-85k: A Language-Conditioned Occupancy Dataset

To train OccDirector with language-conditioned occupancy prediction, we require paired supervision between (i) 4D occupancy representations and (ii) natural-language descriptions that emphasize *geometry, topology, and interaction intent* rather than appearance. To this end, we introduce **OccInteract-85k**, a dataset constructed via an automated pipeline that synthesizes three complementary corpora: *static layout captions* (OccInteract-Static), *environment-interaction clips* (OccInteract-Env), and *agent-interaction simulations* (OccInteract-Agent). All annotations are generated with structured prompting and enforced output schemas, enabling scalable production and automatic filtering.

4.1 Data Curation Pipeline

Our pipeline integrates three data sources to balance realism, scale, and diversity: (i) *Real-world Logs.* We leverage UniOcc [47] to aggregate real-world autonomous driving logs, providing unified occupancy representations with high photorealism. (ii) *Large-scale Simulation.* To overcome the scalability limits of real logs, we collect extensive simulator-based data following the CarlaSC protocol [48], enabling diverse environmental configurations. (iii) *Safety-critical Scenarios.* To target long-tail and safety-critical interactions, which are often absent in real logs. Therefore we generate text-defined scenarios using TTSG [33]. These are rendered via CarlaSC, ensuring coverage of high-risk edge cases.

Standardization. We standardize these raw sources into paired *occupancy–language* training instances. Following UniOcc [47], we standardize all data sources into a unified occupancy grid format. Then, we render semantic occupancy grids into BEV sequences and prompt a frozen VLM (`qwen3-vl-235b-a22b-instruct`) to generate schema-constrained, machine-parseable JSON captions. This process yields annotations at three temporal granularities: (i) single-frame layout captions for learning geometric priors, (ii) clips around key maneuvers for learning behavior-conditioned structure, and (iii) specified multi-agent plans for generating diverse interactions, forming the sub-datasets detailed below. Table 1 summarizes the statistics. Concretely, all sources are unified to a $200\times200\times16$ voxel grid (0.4 m/voxel, X/Y: ±40 m, Z: -3.2–3.2 m) sampled at 2 Hz, with semantic labels remapped to a shared 11-category ontology following UniOcc [47].

4.2 Dataset Composition

OccInteract-Static: Static Layout Priors. This subset pairs single-frame BEV semantic occupancy snapshots with ego-centric layout captions. The annotations focus on road topology (e.g., lane structures, intersections) and drivable regions, providing fundamental geometric priors independent of temporal dynamics.

Table 1: Dataset Statistics. Overview of the processed corpora. Instances denote single frames for OccInteract-Static and video clips for OccInteract-Env / OccInteract-Agent.

Dataset	# Instances	Total Duration
OccInteract-Static	73.5k	N/A (Single-frame)
OccInteract-Env	7.6k	128 h
OccInteract-Agent	4.6k	12 h

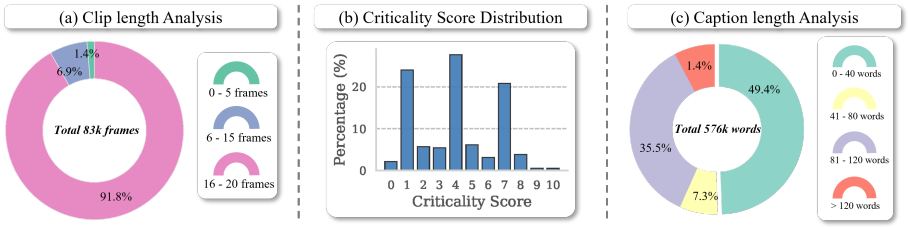


Fig. 3: Data Statistics of OccInteract-Agent. (a) Distribution of clip lengths. (b) Histogram of VLM-predicted criticality scores. (c) Distribution of caption lengths.

OccInteract-Env: Behavior-Conditioned Dynamics. Targeting key maneuvers at junctions, this subset provides clip-level supervision. Each instance includes a discrete ego-action command (straight/left/right) and a description of the local topology. This facilitates learning how occupancy structures evolve conditioned on ego-behavior.

OccInteract-Agent: Multi-Agent Interactions. This corpus covers a spectrum from routine driving to safety-critical near-misses. Each instance is paired with an interaction-focused caption and a *criticality score* ($s \in [0, 10]$), predicted by a frozen VLM judge (`qwen3-vl-235b-a22b-instruct`). We interpret this score as a proxy for risk: a low score corresponds to routine, non-conflicting interactions; medium indicates potential conflicts requiring caution (e.g., nearby vulnerable road users or occlusions); high reflects imminent hazards or near-collision situations. As shown in Fig. 3 (b), the scores exhibit a trimodal distribution (peaks at 1/4/7), effectively stratifying samples into low, medium, and high-risk regimes. This metadata enables curriculum learning and targeted sampling of rare safety-critical scenarios. Furthermore, Fig. 3 (a) and (c) present the statistics for clip durations and caption lengths.

4.3 Automated Quality Assessment

To close the loop between data generation and model learning, we implement a rubric-based VLM evaluator. This mechanism serves a dual purpose: filtering low-quality synthetic data and benchmarking model performance (see Sec. 5).

Table 2: Automated Data Validation. Mean rubric scores (1-5) assessing the quality of the generated dataset.

Dataset	Judge VLM	Completeness ↑	Structural ↑	Sem. Align. ↑	Mean ↑
OccInteract-Static	`gemini-3.0-pro`	4.16	4.20	3.84	4.08
	`gpt-5.2`	2.97	3.71	3.03	3.24
	`qwen3-vl-235b-a22b-instruct`	3.22	3.84	3.97	3.68
OccInteract-Env	`gemini-3.0-pro`	4.62	4.60	3.55	4.26
	`gpt-5.2`	3.21	3.78	3.01	3.33
	`qwen3-vl-235b-a22b-instruct`	3.32	3.81	3.66	3.60
OccInteract-Agent	`gemini-3.0-pro`	4.51	4.53	3.49	4.18
	`gpt-5.2`	3.21	3.72	2.58	3.17
	`qwen3-vl-235b-a22b-instruct`	3.30	3.64	3.32	3.42

Evaluation Protocol. We adopt a reference-free evaluation paradigm. Specifically, we render the generated semantic occupancy grids into BEV video clips and prompt a frozen VLM judge to score them against the source text descriptions. We define three evaluation axes, scored on an integer scale of $[1, 5]$: (i) *scene completeness* (coverage of relevant static/dynamic elements), (ii) *structural correctness* (physical plausibility and topological consistency), and (iii) *Semantic Alignment* (faithfulness to the prompt).

To ensure robustness, we employ three state-of-the-art VLMs as judges: `gemini-3.0-pro`, `gpt-5.2`, and `qwen3-vl-235b-a22b-instruct`. We sample 100 instances from each sub-dataset, with each instance evaluated five times to reduce variance.

Analysis. Table 2 presents a comprehensive evaluation across three axes. First, regarding *absolute quality*, `gemini-3.0-pro` consistently awards high scores (mean > 4.0), confirming that our synthesized occupancy scenes possess high completeness and physical plausibility. Second, regarding *scorer calibration*, we observe a systematic offset between judges; for instance, `gpt-5.2` is notably stricter (mean ≈ 3.2). We attribute this to the inherent visual abstraction of voxelized representations, which stricter VLMs may penalize compared to photorealistic imagery. Third, and most importantly, we demonstrate *pipeline robustness* through relative consistency. Despite the increasing complexity from static layouts (OccInteract-Static) to multi-agent interactions (OccInteract-Agent), the score variance under the same judge remains minimal (e.g., `gpt-5.2` means range tightly: 3.17–3.33). This stability indicates that our automated pipeline maintains uniform fidelity regardless of scenario complexity, validating the reliability of OccInteract-85k for large-scale training.

5 Experiments

5.1 Experimental Setup

Data. We evaluate on the same processed dataset described in Sec. 4, which contains about 85k training instances.

Compared methods. We compare OccDirector with recent state-of-the-art 4D occupancy generation methods, including DOME [14], COME [36], and DynamicCity [3]. As discussed in Sec. 1, these traditional approaches rely on explicit geometric conditions, such as predefined ego-vehicle trajectories or semantic layouts, to control the generation process. To the best of our knowledge, OccDirector is the pioneering framework for purely text-guided 4D occupancy generation. Lacking direct off-the-shelf baselines, we adapt DynamicCity by replacing its geometric control tokens with text embeddings from pre-trained encoders, establishing two baselines: *DynamicCity + CLIP [31]* and *DynamicCity + T5 [32]*. Finally, we report results for two variants of our proposed model: (i) w/o history-prefix control, and (ii) w/ history-prefix control.

Implementation Details. We implement OccDirector using PyTorch, trained for 100k iterations with batch size 32, AdamW optimizer (lr=1×10^{-4}, β_1=0.9, β_2=0.95), and CFG dropout probability 0.15. The MMDiT backbone has 14 transformer layers (d_{model}=896) with a 2-layer Token Refiner. At inference we use 20 Euler steps with CFG scale 7.5.

Evaluation Metrics. Following previous work [3, 21, 54], we render the 3D/4D occupancy grids into BEV-view RGB images and clips. These rendered RGB representations are subsequently utilized to evaluate all single-frame and video-level visual quality metrics. Specifically, we report Inception Score (IS) [2], Fréchet Inception Distance (FID) [15], Kernel Inception Distance (KID) [4] and Fréchet Video Distance (FVD) [42] to assess fidelity and temporal coherence, alongside Precision and Recall for diversity. For instruction alignment, we employ the VLM-based instruction alignment score (Sec. 4.3). To ensure consistent and rigorous comparison, we fix `gemini-3.0-pro` as our sole judge VLM for all experiments. For evaluation, we uniformly sample 10K ground-truth 16-frame clips and generate a matching set via the CFG unconditional branch. Single-frame metrics (IS, FID, KID, Precision, Recall) are computed on 5 frames uniformly sampled per clip using Inception-v3 [38] features; video-level FVD and video Precision/Recall use the full 16-frame sequences with I3D [5] features.

5.2 Generation Quality Results

Quantitative results. We present the single-frame visual quality results in Tab. 3 and the video generation quality results in Tab. 4.

As shown in Tab. 3, OccDirector demonstrates highly competitive per-frame generation quality. It is important to note that traditional methods (e.g., DOME, COME, DynamicCity) rely on explicit geometric conditions (e.g., trajectories or layouts) which naturally constrain the generation space and aid in structural consistency. Despite OccDirector being designed for the more abstract text-to-4D task, our unconditional samples achieve an Inception Score (IS) of 3.40 and FID of 62, performance that is comparable to these geometric-guided upper bounds. Crucially, when compared to the direct text-adapted baselines (DynamicCity + CLIP/T5) which share the same functional goal, our method consistently

Table 3: Single-frame visual quality. We evaluate per-frame image fidelity and diversity using Inception Score (IS), Fréchet Inception Distance (FID), Kernel Inception Distance (KID), Precision and Recall.

Method	IS ↑	FID ↓	KID ↓	Precision ↑	Recall ↑
DOME [14]	3.21	67	0.06	0.38	**0.23**
COME [36]	3.26	77	0.06	**0.40**	**0.23**
DynamicCity [3]	**3.39**	**61**	**0.05**	0.31	**0.23**
DynamicCity + CLIP	3.39	65	**0.05**	0.32	**0.24**
DynamicCity + T5	3.36	66	**0.05**	0.31	0.22
OccDirector w/o history-prefix control	3.39	64	**0.05**	**0.34**	**0.24**
OccDirector w/ history-prefix control	**3.40**	**62**	**0.05**	0.33	0.23

Table 4: Video generation quality and instruction alignment. We evaluate video-level fidelity and diversity using Fréchet Video Distance (FVD), Precision and Recall. Instruction alignment is evaluated using the VLM-based rubric score (Sec. 4.3).

	Video realism			Instruction alignment			
Method	FVD ↓	Precision ↑	Recall ↑	Completeness ↑	Structural ↑	Sem. Align. ↑	Mean ↑
DOME [14]	437	0.55	0.03	N/A	N/A	N/A	N/A
COME [36]	**391**	0.56	0.04	N/A	N/A	N/A	N/A
DynamicCity [3]	401	**0.57**	**0.08**	N/A	N/A	N/A	N/A
DynamicCity + CLIP	388	0.60	0.008	3.03	3.54	2.69	3.09
DynamicCity + T5	396	**0.64**	0.02	2.84	3.34	2.73	2.97
OccDirector w/o history-prefix control	354	0.48	**0.12**	3.59	3.66	3.04	3.43
OccDirector w/ history-prefix control	**275**	0.58	**0.12**	**3.81**	**4.01**	**3.32**	**3.73**

achieves superior performance across all single-frame metrics. This confirms that our architecture effectively learns the underlying scene distribution and generates high-fidelity occupancy frames, outperforming naive adaptations of previous works.

For video-level generation (Tab. 4, left), OccDirector significantly improves temporal coherence and dynamic realism. Our full model achieves an FVD of 275, outperforming the strongest baseline (DynamicCity + CLIP, FVD 388) by a remarkable 29.1%. Furthermore, OccDirector achieves the highest video Recall (0.12) while maintaining a strong Precision (0.58). We observe an interesting phenomenon with the text-adapted baselines (DynamicCity + CLIP/T5): while they achieve relatively high Precision (e.g., 0.64 for T5), their Recall is extremely low (e.g., 0.008 for CLIP). This suggests a severe mode collapse, where standard text encoders fail to adequately capture complex text instructions, leading the model to repeatedly generate a limited set of "safe" or static scenarios. In contrast, our method generates spatio-temporal occupancy sequences that are not only realistic but also encompass a significantly wider diversity of dynamic scenarios.

Furthermore, in terms of text-to-occupancy instruction alignment (Tab. 4, right), OccDirector substantially surpasses the text-adapted baselines across all VLM-based rubric scores. For instance, our full model achieves a mean

score of 3.73. The low semantic alignment scores of DynamicCity + CLIP/T5 further corroborate our observation regarding their low Recall: standard text encoders struggle to differentiate between diverse prompts, resulting in poor alignment and limited generation diversity. This highlights the severe semantic-spatiotemporal gap in standard text encoders and proves the superiority of our VLM-driven architecture. Notably, the inclusion of the history-prefix control mechanism further boosts the structural correctness and instruction-following capabilities (improving the mean score from 3.43 to 3.73), demonstrating the effectiveness of our proposed conditioning design for complex multi-agent scenario orchestration.

Qualitative results on history-prefix control. As the teaser in Sec. 1 highlights OccDirector's diverse controllability, we further investigate its capability in long-horizon autoregressive rollouts. Fig. 4 illustrates two multi-stage narratives where the ego vehicle is denoted by a *black car icon*. In the **stop-and-go scenario** (top), the model follows the instruction to decelerate on a one-way street and stop behind a van at a red light (Stage 1). In Stage 2, responding to the "green light" prompt, the van initiates movement across the intersection edge (marked by *red dashed lines*), correctly prompting the ego to prepare to follow. In the **emergency braking scenario** (bottom), the ego initially cruises on a wide, empty straight road. Subsequently, the model successfully generates a sudden pedestrian crossing event at a T-intersection (highlighted by a *red circle*), forcing the ego to execute an emergency stop. These results confirm that history-prefix control effectively maintains spatio-temporal coherence and causal logic across generation stages.

Qualitative comparison on fine-grained text alignment. Fig. 5 compares OccDirector with text-adapted baselines on three prompts requiring precise semantic understanding. In the first column "...follows a lead car, with *no other* traffic...", baselines fail to respect the negation, generating extraneous vehicles. We highlight these hallucinations in baselines versus the correct single lead car in our result using *red circles*. In the second column "...moving ego... *stop* by a bus", baselines exhibit attribute confusion (e.g., DynamicCity + CLIP renders a completely static scene, while the T5 variant incorrectly halts the ego vehicle while moving the bus). Here, *red circles* contrast the baselines' incorrect objects with our successfully generated blocking bus. Finally, for the "cut-in" prompt, we indicate the ego's forward direction with *red dashed lines*. While baselines revert to simple lane-following, only OccDirector synthesizes the lead car's lateral intrusion across this line, demonstrating superior grasp of spatial dynamics.

5.3 Ablation Study

To thoroughly evaluate the effectiveness of our proposed architectural designs, we conduct comprehensive ablation studies. Since the impact of the history-prefix control has already been demonstrated in Tab. 4 and Fig. 4, we establish our ablation baseline using the text-to-occupancy generation setting **without (w/o)**

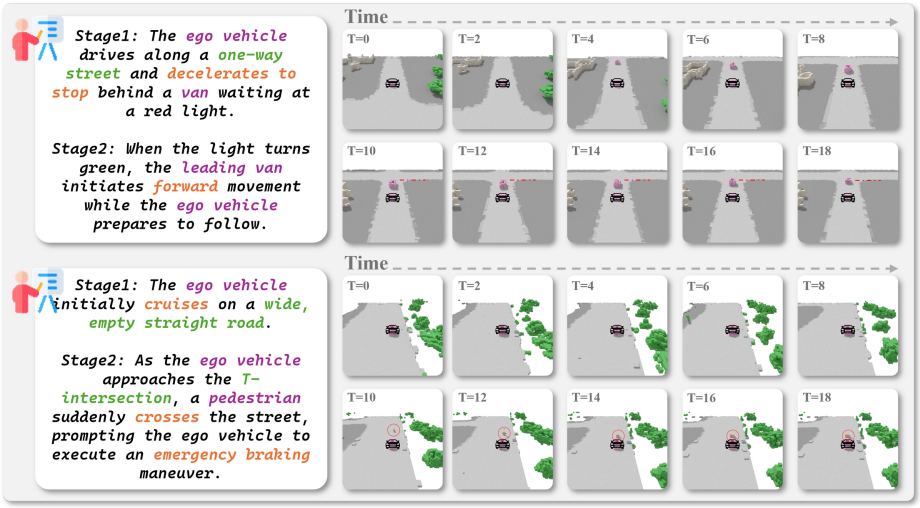


Fig. 4: Long-horizon narrative generation. We visualize multi-stage rollouts for a stop-and-go scenario (top) and an emergency braking event (bottom).

Table 5: Ablation study on model architecture. We evaluate the impact of the attention mechanism (Full vs. STSA) and the Token Refiner under the setting w/o history control.

Architecture Design		Video realism			VLM score ↑			
Backbone	Token Refiner	FVD ↓	Precision ↑	Recall ↑	Completeness ↑	Structural ↑	Instr. follow ↑	Mean ↑
Full attention	×	592	**0.48**	0.02	**2.98**	**2.86**	2.48	2.77
Full attention	✓	**532**	0.41	**0.07**	2.97	2.74	**2.63**	**2.78**
STSA (ours)	×	407	0.49	**0.14**	3.46	3.39	2.86	3.24
STSA (ours)	✓	**354**	**0.58**	0.12	**3.59**	**3.66**	**3.04**	**3.43**

history control. We focus on two key components: (i) The backbone strategy, where we compare our Spatio-Temporal Separated Attention (STSA) against a Full Attention baseline that applies standard joint attention over the entire flattened 4D token sequence. (ii) The Token Refiner, a bidirectional transformer module designed to bridge the semantic-spatial gap of VLM features. Results are summarized in Tab. 5.

Effectiveness of STSA and Token Refiner. First, STSA is critical for modeling 4D dynamics. Compared to the Full Attention baseline (FVD 532), STSA significantly reduces FVD to 354 and improves the VLM score (2.78 → 3.43), confirming the benefit of factorizing spatial and temporal computation. Second, the Token Refiner effectively bridges the gap between frozen text embeddings and occupancy features. Removing the refiner degrades performance (FVD 354 → 407), verifying its necessity for aligning high-dimensional text features with spatial

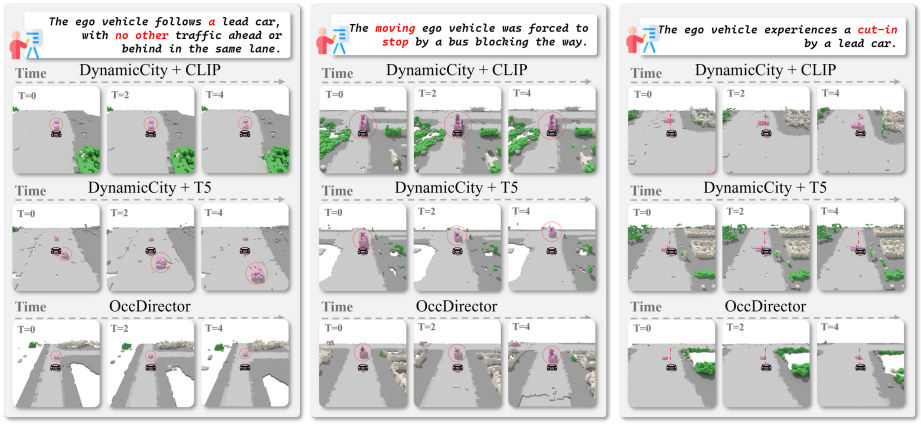


Fig. 5: Qualitative comparison of text-to-occupancy alignment. We evaluate adherence to fine-grained instructions: (Left) Negation and cardinality; (Middle) Dynamic attribute binding; (Right) Complex "cut-in" trajectory.

constraints. Notably, the Full Attention backbone fails to fully leverage the refined tokens (VLM score stagnates at 2.78), whereas our STSA architecture successfully integrates them to achieve state-of-the-art fidelity and alignment.

6 Conclusion

In this paper, we presented OccDirector, a pioneering framework capable of generating dynamic 4D occupancy scenes purely from natural language instructions. By shifting the paradigm from rigid geometric conditioning to high-level semantic orchestration, OccDirector acts as a "scenario director", enabling intuitive control over complex multi-agent behaviors and environmental interactions. To achieve this, we introduced a VLM-driven Spatio-Temporal MMDiT architecture equipped with a novel history-prefix anchoring strategy, effectively bridging the semantic-spatiotemporal gap and ensuring physically plausible, interaction-consistent rollouts. Furthermore, we contributed OccInteract-85k, a large-scale dataset with multi-level language descriptions, alongside a rigorous VLM-based evaluation benchmark. Extensive experiments demonstrate that our approach achieves state-of-the-art generation quality and unprecedented instruction-following capabilities.

References

1. Alaba, S.Y., Ball, J.E.: A survey on deep-learning-based lidar 3d object detection for autonomous driving. Sensors **22**(24), 9577 (2022)
2. Barratt, S., Sharma, R.: A note on the inception score. arXiv preprint arXiv:1801.01973 (2018)
3. Bian, H., Kong, L., Xie, H., Pan, L., Qiao, Y., Liu, Z.: Dynamiccity: Large-scale 4d occupancy generation from dynamic scenes. arXiv preprint arXiv:2410.18084 (2024)
4. Bińkowski, M., Sutherland, D.J., Arbel, M., Gretton, A.: Demystifying mmd gans. arXiv preprint arXiv:1801.01401 (2018)
5. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
6. Chen, L., Wu, P., Chitta, K., Jaeger, B., Geiger, A., Li, H.: End-to-end autonomous driving: Challenges and frontiers. IEEE Transactions on Pattern Analysis and Machine Intelligence **46**(12), 10164–10183 (2024)
7. Chen, R., Wu, Z., Liu, Y., Guo, Y., Ni, J., Xia, H., Xia, S.: Unimlvg: Unified framework for multi-view long video generation with comprehensive control capabilities for autonomous driving. arXiv preprint arXiv:2412.04842 (2024)
8. Diehl, C., Sykora, Q., Agro, B., Gilles, T., Casas, S., Urtasun, R.: Dio: Decomposable implicit 4d occupancy-flow world model. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 27456–27466 (2025)
9. Dong, W., Lu, S., Chen, X., Zhang, S., Liu, Q., Liu, Z., Chen, L., Wang, H., Cai, Y.: End-to-end autonomous driving: From classic paradigm to large model empowerment—a comprehensive survey. IEEE Internet of Things Journal **13**(3), 3870–3898 (2025)
10. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first international conference on machine learning (2024)
11. Gao, R., Chen, K., Xiao, B., Hong, L., Li, Z., Xu, Q.: Magicdrive-v2: High-resolution long video generation for autonomous driving with adaptive control. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 28135–28144 (2025)
12. Gao, R., Chen, K., Xie, E., Hong, L., Li, Z., Yeung, D.Y., Xu, Q.: Magicdrive: Street view generation with diverse 3d geometry control. arXiv preprint arXiv:2310.02601 (2023)
13. Gao, S., Yang, J., Chen, L., Chitta, K., Qiu, Y., Geiger, A., Zhang, J., Li, H.: Vista: A generalizable driving world model with high fidelity and versatile controllability. Advances in Neural Information Processing Systems **37**, 91560–91596 (2024)
14. Gu, S., Yin, W., Jin, B., Guo, X., Wang, J., Li, H., Zhang, Q., Long, X.: Dome: Taming diffusion model into high-fidelity controllable occupancy world model. arXiv preprint arXiv:2410.10429 (2024)
15. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems **30** (2017)
16. Hu, A., Russell, L., Yeo, H., Murez, Z., Fedoseev, G., Kendall, A., Shotton, J., Corrado, G.: Gaia-1: A generative world model for autonomous driving. arXiv preprint arXiv:2309.17080 (2023)

17. Khurana, T., Hu, P., Held, D., Ramanan, D.: Point cloud forecasting as a proxy for 4d occupancy forecasting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1116–1124 (2023)
18. Kim, S.W., Philion, J., Torralba, A., Fidler, S.: Drivegan: Towards a controllable high-quality neural simulation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5820–5829 (2021)
19. Koishigarina, D., Uselis, A., Oh, S.J.: Clip behaves like a bag-of-words model cross-modally but not uni-modally. arXiv preprint arXiv:2502.03566 (2025)
20. Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., Xiong, J., Li, X., Wu, B., Zhang, J., et al.: Hunyuanvideo: A systematic framework for large video generative models. arXiv preprint arXiv:2412.03603 (2024)
21. Lee, J., Lee, S., Jo, C., Im, W., Seon, J., Yoon, S.E.: Semcity: Semantic scene generation with triplane diffusion. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 28337–28347 (2024)
22. Li, B., Guo, J., Liu, H., Zou, Y., Ding, Y., Chen, X., Zhu, H., Tan, F., Zhang, C., Wang, T., et al.: Uniscene: Unified occupancy-centric driving scene generation. In: Proceedings of the computer vision and pattern recognition conference. pp. 11971–11981 (2025)
23. Li, L., Shao, W., Dong, W., Tian, Y., Zhang, Q., Yang, K., Zhang, W.: Data-centric evolution in autonomous driving: A comprehensive survey of big data system, data mining, and closed-loop technologies. arXiv preprint arXiv:2401.12888 (2024)
24. Liang, A., Kong, L., Yan, T., Liu, H., Yang, W., Huang, Z., Yin, W., Zuo, J., Hu, Y., Zhu, D., et al.: Worldlens: Full-spectrum evaluations of driving world models in real world. arXiv preprint arXiv:2512.10958 (2025)
25. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. arXiv preprint arXiv:2210.02747 (2022)
26. Liu, Y., Zhang, K., Li, Y., Yan, Z., Gao, C., Chen, R., Yuan, Z., Huang, Y., Sun, H., Gao, J., et al.: Sora: A review on background, technology, limitations, and opportunities of large vision models. arXiv preprint arXiv:2402.17177 (2024)
27. Ma, B., Zong, Z., Song, G., Li, H., Liu, Y.: Exploring the role of large language models in prompt encoding for diffusion models. arXiv preprint arXiv:2406.11831 (2024)
28. Ma, J., Chen, X., Huang, J., Xu, J., Luo, Z., Xu, J., Gu, W., Ai, R., Wang, H.: Cam4docc: Benchmark for camera-only 4d occupancy forecasting in autonomous driving applications. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21486–21495 (2024)
29. Ma, X., Wang, Y., Chen, X., Jia, G., Liu, Z., Li, Y.F., Chen, C., Qiao, Y.: Latte: Latent diffusion transformer for video generation. arXiv preprint arXiv:2401.03048 (2024)
30. Mei, J., Hu, T., Yang, X., Wen, L., Yang, Y., Wei, T., Ma, Y., Dou, M., Shi, B., Liu, Y.: Dreamforge: Motion-aware autoregressive video generation for multi-view driving scenes. arXiv preprint arXiv:2409.04003 (2024)
31. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021)
32. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of machine learning research **21**(140), 1–67 (2020)

33. Ruan, B.K., Tsui, H.T., Li, Y.H., Shuai, H.H.: Traffic scene generation from natural language description for autonomous vehicles with large language model. arXiv preprint arXiv:2409.09575 (2024)
34. Russell, L., Hu, A., Bertoni, L., Fedoseev, G., Shotton, J., Arani, E., Corrado, G.: Gaia-2: A controllable multi-view generative world model for autonomous driving. arXiv preprint arXiv:2503.20523 (2025)
35. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. Advances in neural information processing systems **35**, 36479–36494 (2022)
36. Shi, Y., Jiang, K., Meng, Q., Wang, K., Wang, J., Sun, W., Wen, T., Yang, M., Yang, D.: Come: Adding scene-centric forecasting control to occupancy world model. arXiv preprint arXiv:2506.13260 (2025)
37. Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding. Neurocomputing **568**, 127063 (2024)
38. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016)
39. Tian, X., Jiang, T., Yun, L., Mao, Y., Yang, H., Wang, Y., Wang, Y., Zhao, H.: Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. Advances in Neural Information Processing Systems **36**, 64318–64330 (2023)
40. Tong, S., Fan, D., Li, J., Xiong, Y., Chen, X., Sinha, K., Rabbat, M., LeCun, Y., Xie, S., Liu, Z.: Metamorph: Multimodal understanding and generation via instruction tuning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17001–17012 (2025)
41. Tong, W., Sima, C., Wang, T., Chen, L., Wu, S., Deng, H., Gu, Y., Lu, L., Luo, P., Lin, D., et al.: Scene as occupancy. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8406–8415 (2023)
42. Unterthiner, T., Van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., Gelly, S.: Towards accurate generative models of video: A new metric & challenges. arXiv preprint arXiv:1812.01717 (2018)
43. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
44. Wang, L., Zheng, W., Ren, Y., Jiang, H., Cui, Z., Yu, H., Lu, J.: Occsora: 4d occupancy generation models as world simulators for autonomous driving. arXiv preprint arXiv:2405.20337 (2024)
45. Wang, X., Zhu, Z., Huang, G., Chen, X., Zhu, J., Lu, J.: Drivedreamer: Towards real-world-drive world models for autonomous driving. In: European conference on computer vision. pp. 55–72. Springer (2024)
46. Wang, X., Zhu, Z., Xu, W., Zhang, Y., Wei, Y., Chi, X., Ye, Y., Du, D., Lu, J., Wang, X.: Openoccupancy: A large scale benchmark for surrounding semantic occupancy perception. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17850–17859 (2023)
47. Wang, Y., Huang, X., Sun, X., Yan, M., Xing, S., Tu, Z., Li, J.: Uniocc: A unified benchmark for occupancy forecasting and prediction in autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 25560–25570 (2025)
48. Wilson, J., Song, J., Fu, Y., Zhang, A., Capodieci, A., Jayakumar, P., Barton, K., Ghaffari, M.: Motionsc: Data set and network for real-time semantic mapping in

dynamic environments. IEEE Robotics and Automation Letters **7**(3), 8439–8446 (2022)
49. Wu, C., Li, J., Zhou, J., Lin, J., Gao, K., Yan, K., Yin, S.m., Bai, S., Xu, X., Chen, Y., et al.: Qwen-image technical report. arXiv preprint arXiv:2508.02324 (2025)
50. Wu, C., Chen, X., Wu, Z., Ma, Y., Liu, X., Pan, Z., Liu, W., Xie, Z., Yu, X., Ruan, C., et al.: Janus: Decoupling visual encoding for unified multimodal understanding and generation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 12966–12977 (2025)
51. Xu, H., Chen, J., Meng, S., Wang, Y., Chau, L.P.: A survey on occupancy perception for autonomous driving: The information fusion perspective. Information Fusion **114**, 102671 (2025)
52. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al.: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025)
53. Yang, L., Tian, Y., Li, B., Zhang, X., Shen, K., Tong, Y., Wang, M.: Mmada: Multimodal large diffusion language models. arXiv preprint arXiv:2505.15809 (2025)
54. Yang, Y., Liang, A., Mei, J., Ma, Y., Liu, Y., Lee, G.H.: X-scene: Large-scale driving scene generation with high fidelity and flexible controllability. arXiv preprint arXiv:2506.13558 (2025)
55. Yang, Z., Liu, Z., Lu, Y., Hou, L., Miao, C., Peng, S., Feng, B., Bai, X., Zhao, H.: Geniedrive: Towards physics-aware driving world model with 4d occupancy guided video generation. arXiv preprint arXiv:2512.12751 (2025)
56. Yuksekgonul, M., Bianchi, F., Kalluri, P., Jurafsky, D., Zou, J.: When and why vision-language models behave like bags-of-words, and what to do about it? arXiv preprint arXiv:2210.01936 (2022)
57. Zhang, Y., Zhu, Z., Du, D.: Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9433–9443 (2023)
58. Zhao, G., Wang, X., Zhu, Z., Chen, X., Huang, G., Bao, X., Wang, X.: Drivedreamer-2: Llm-enhanced world models for diverse driving video generation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 10412–10420 (2025)
59. Zheng, W., Chen, W., Huang, Y., Zhang, B., Duan, Y., Lu, J.: Occworld: Learning a 3d occupancy world model for autonomous driving. In: European conference on computer vision. pp. 55–72. Springer (2024)

OccDirector: Language-Guided Behavior and Interaction Generation in 4D Occupancy Space

Supplementary Material

Zhuding Liang[1⋆], Tianyi Yan[1*], Dubing Chen[1], Jiasen Zheng[2], Huan Zheng[1], Cheng-zhong Xu[1], Yida Wang[3], Kun Zhan[3], and Jianbing Shen[1⋆⋆]

[1] SKL-IOTSC, CIS, University of Macau, Macau, China
[2] Northwestern University, Evanston, United States
[3] Li Auto Inc, Beijing, China

Table of Contents

A Additional Implementation Details . . . 21
A.1 OccDirector Components and Hyperparameters . . . 21
A.2 OccInteract-85k Standardization and Annotation Details . . . 24
A.3 Prompt Templates for Automated Quality Assessment . . . 30
A.4 Evaluation Protocol and Metric Definitions . . . 33
B Additional Quantitative Results . . . 34
B.1 Inference Resource Consumption . . . 34
B.2 Architectural Sensitivity . . . 34
B.3 Methodological Ablations . . . 35
B.4 Robustness of VLM-based Evaluation . . . 35
B.5 User Study . . . 36
C Additional Qualitative Results . . . 36
C.1 Dataset Samples and Criticality Analysis . . . 37
C.2 Hierarchical Generation . . . 38
C.3 History-Prefix Control . . . 38
C.4 Fine-grained Text-to-Occupancy Alignment . . . 40
D Potential Societal Impact & Limitations . . . 41
D.1 Potential Societal Impact . . . 41
D.2 Limitations . . . 42
E Public Resources Used . . . 42
E.1 Data Source & Privacy Declaration . . . 42
E.2 Public Models & Benchmarking Vision-Language Models . . . 43
E.3 Public Implementations & Libraries . . . 43

⋆ Equal Contribution.
⋆⋆ Corresponding Author.

A Additional Implementation Details

A.1 OccDirector Components and Hyperparameters

1. Joint Latent Representation. We formalize the derivation of the occupancy tokens $\mathbf{H}$ and the text tokens $\mathbf{C}$.

Occupancy Tokenization: Given a discrete 4D occupancy rollout $\mathbf{O}_{1:F} \in \{0, \dots, K-1\}^{F\times 200\times 200\times 16}$, we first linearly interpolate the spatial dimensions to $F \times 224 \times 224 \times 16$ to ensure compatibility with standard 2×2 patchification. The frozen Occ-VAE maps each occupancy voxel ID to a dense embedding via a pre-trained lookup table $\mathrm{Emb}(\cdot)$ with expansion ratio $e = 8$:

$$\mathbf{E} = \mathrm{Emb}(\mathbf{O}_{1:F}) \in \mathbb{R}^{F\times 224\times 224\times 16\times 8}. \tag{S1}$$

To achieve the efficient "3D spatial tokens" representation, we fold the depth/height bins $D = 16$ into the channel axis, obtaining a channel-first latent $\mathbf{X} \in \mathbb{R}^{128\times F\times 224\times 224}$. This high-dimensional spatial representation is then compressed by the Occ-VAE encoder to a continuous spatio-temporal latent grid $\mathbf{z} \in \mathbb{R}^{16\times F\times 28\times 28}$ via reparameterization:

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{S2}$$

To interface with the transformer, we apply a spatial patchification (Unfold) with a patch size $p = 2$, resulting in $S = (28/2)^2 = 196$ spatial tokens per frame. These tokens are projected to the model dimension $d_{\text{model}} = 896$ via a linear layer.

Text Token Refinement: We utilize Qwen3-VL-2B [14] as OccDirector's frozen VLM. It extracts raw text features $\tilde{\mathbf{C}} \in \mathbb{R}^{L\times d_{\text{text}}}$ from the prompt π. The Token Refiner $\mathrm{Refine}(\cdot)$ is a lightweight bidirectional transformer consisting of N_{ref} blocks. Let $\mathbf{C}^{(0)} = \tilde{\mathbf{C}}\mathbf{W}_{\text{in}}$ be the initial projection to d_{model}. Each block $i \in \{1, \dots, N_{\text{ref}}\}$ performs:

$$\hat{\mathbf{C}}^{(i)} = \mathbf{C}^{(i-1)} + \mathrm{MHSA}(\mathrm{LN}(\mathbf{C}^{(i-1)})), \tag{S3}$$

$$\mathbf{C}^{(i)} = \hat{\mathbf{C}}^{(i)} + \mathrm{MLP}(\mathrm{LN}(\hat{\mathbf{C}}^{(i)})), \tag{S4}$$

where MHSA is an unmasked, bidirectional multi-head self-attention to capture global semantic relations. The final refined text tokens are $\mathbf{C} = \mathbf{C}^{(N_{\text{ref}})} \in \mathbb{R}^{L\times d_{\text{model}}}$.

2. Scalable 4D Generation Backbone. We detail the internal structure of the l-th transformer block. The input consists of the packed occupancy tokens $\mathbf{H}^{(l-1)} \in \mathbb{R}^{F\times S\times d_{\text{model}}}$ and the text tokens $\mathbf{C}^{(l-1)} \in \mathbb{R}^{L\times d_{\text{model}}}$. The data flow proceeds through four ordered stages.

(I) Pre-Modulation. Given the integration timestep t, we compute the timestep embedding $\mathbf{s}(t)$ via sinusoidal positional encoding [10] followed by an MLP. For each modality stream $m \in \{\text{occ}, \text{txt}\}$, a learned MLP predicts two sets of modulation parameters:

$$(\boldsymbol{\alpha}_{m,1}, \boldsymbol{\beta}_{m,1}, \mathbf{g}_{m,1}, \boldsymbol{\alpha}_{m,2}, \boldsymbol{\beta}_{m,2}, \mathbf{g}_{m,2}) = \mathrm{MLP}_m^{(l)}(\mathbf{s}(t)). \tag{S5}$$

We apply AdaLN to prepare the queries, keys, and values for the attention mechanism:

$$\hat{\mathbf{X}}_m = \mathrm{LN}(\mathbf{X}_m^{(l-1)}) \odot (1 + \boldsymbol{\alpha}_{m,1}) + \boldsymbol{\beta}_{m,1}, \quad \text{where } \mathbf{X} \in \{\mathbf{H}, \mathbf{C}\}. \tag{S6}$$

(II) Spatial Joint Attention (Dual-Stream). We first model the spatial relationships and text-to-geometry grounding. The modulated occupancy sequence is reshaped into frame-wise batches $\hat{\mathbf{H}}_f \in \mathbb{R}^{S \times d_{\text{model}}}$. To encode spatial geometry, we apply 2D Rotary Positional Embeddings (RoPE) to the queries and keys of $\hat{\mathbf{H}}_f$ based on their 2D spatial grid coordinates (h, w). *Implementation Note:* Our model conceptually operates on 3D spatial tokens. In our efficient implementation, the Z-axis (height) is densely folded into the channel dimension. Thus, applying 2D RoPE on the $H \times W$ grid mathematically serves as the spatial component of the 3D+1D RoPE, preserving full 3D geometric awareness without the cubic complexity of explicit 3D attention.

For each frame f, we concatenate the shared text tokens $\hat{\mathbf{C}}$ and the frame-specific occupancy tokens $\hat{\mathbf{H}}_f$ to form a joint sequence. We perform self-attention on this unified sequence to capture cross-modal interactions:

$$[\Delta\mathbf{C}_f; \Delta\mathbf{H}_f] = \mathrm{Attention}([\hat{\mathbf{C}}; \hat{\mathbf{H}}_f]), \tag{S7}$$

where the output is split back into the text update $\Delta\mathbf{C}_f \in \mathbb{R}^{L \times d_{\text{model}}}$ and the spatial occupancy update $\Delta\mathbf{H}_f \in \mathbb{R}^{S \times d_{\text{model}}}$.

The streams are then updated via residual connections gated by $\mathbf{g}_{m,1}$:

$$\mathbf{H}' = \mathbf{H}^{(l-1)} + \mathbf{g}_{\text{occ},1} \odot \mathrm{Concat}(\Delta\mathbf{H}_1, \ldots, \Delta\mathbf{H}_F), \tag{S8}$$

$$\mathbf{C}' = \mathbf{C}^{(l-1)} + \mathbf{g}_{\text{txt},1} \odot \frac{1}{F} \sum_{f=1}^{F} \Delta\mathbf{C}_f. \tag{S9}$$

(III) Temporal Self-Attention (Image Only). Following the spatial interaction, we model the temporal dynamics exclusively on the occupancy stream. We reshape the intermediate tokens $\mathbf{H}'$ to isolate temporal tubes $\mathbf{H}'_s \in \mathbb{R}^{F \times d_{\text{model}}}$ for each spatial token $s \in \{1, \ldots, S\}$. We inject temporal order via 1D Temporal RoPE based on the frame index f. The sequence is updated via a temporal self-attention layer, scaled by a learnable scalar gate γ_{tmp}:

$$\mathbf{H}'' = \mathbf{H}' + \gamma_{\text{tmp}} \cdot \mathrm{MHSA}_{\text{tmp}}(\mathbf{H}'_s). \tag{S10}$$

(IV) Post-Modulation and MLP. Finally, we process the tokens through the feed-forward networks. We apply the second set of modulation parameters to the updated tokens $\mathbf{H}''$ and $\mathbf{C}'$:

$$\tilde{\mathbf{H}} = \mathrm{LN}(\mathbf{H}'') \odot (1 + \boldsymbol{\alpha}_{\text{occ},2}) + \boldsymbol{\beta}_{\text{occ},2}, \tag{S11}$$

$$\tilde{\mathbf{C}} = \mathrm{LN}(\mathbf{C}') \odot (1 + \boldsymbol{\alpha}_{\text{txt},2}) + \boldsymbol{\beta}_{\text{txt},2}. \tag{S12}$$

The outputs of the transformer block are obtained by adding the MLP residuals, gated by $\mathbf{g}_{m,2}$:

$$\mathbf{H}^{(l)} = \mathbf{H}'' + \mathbf{g}_{\text{occ},2} \odot \text{MLP}_{\text{occ}}(\tilde{\mathbf{H}}), \tag{S13}$$

$$\mathbf{C}^{(l)} = \mathbf{C}' + \mathbf{g}_{\text{txt},2} \odot \text{MLP}_{\text{txt}}(\tilde{\mathbf{C}}). \tag{S14}$$

3. Interaction-Consistent Forecasting via History Anchoring. Let $\mathbf{x}_0 \in \mathbb{R}^{F\times S\times d_{\text{model}}}$ denote the ground-truth clean latent sequence, and $\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1$ denote the noisy sequence at flow step $t \in [0,1]$. Given a history horizon of h frames, we define a temporal token replacement operator $\mathcal{R}_{\text{rep}}(\cdot,\cdot)$. For a target sequence $\mathbf{x}_t$ and a clean reference $\mathbf{x}_0$, the operator yields a hybrid sequence $\tilde{\mathbf{x}}_t = \mathcal{R}_{\text{rep}}(\mathbf{x}_t, \mathbf{x}_0)$, where for each frame index $f \in \{1,\dots,F\}$:

$$\tilde{\mathbf{x}}_{t,f} = \begin{cases} \mathbf{x}_{0,f}, & \text{if } f \le h \text{ (Clean History Anchor at } t=0), \\ \mathbf{x}_{t,f}, & \text{if } f > h \text{ (Noisy Future to Predict).} \end{cases} \tag{S15}$$

This operator explicitly anchors the conditioning segment at the *clean timestep* ($t = 0$), ensuring the history portion remains uncorrupted and provides a stable spatial-temporal reference.

Training Dynamics: Independent Anchoring and CFG. During training, we employ two independent stochastic dropping mechanisms to enable flexible conditioning:

- *History Anchoring:* With a probability $p_{\text{anchor}} = 0.5$, we apply the replacement operator to construct the partially clean input $\tilde{\mathbf{x}}_t = \mathcal{R}_{\text{rep}}(\mathbf{x}_t, \mathbf{x}_0)$. Otherwise, the model receives the fully noisy sequence $\mathbf{x}_t$. The loss is only computed on the future frames $(h+1:F)$.
- *Classifier-Free Guidance (CFG):* Independently of the anchoring drop, we apply CFG with a probability $p_{\text{cfg}} = 0.15$. When triggered, the refined text tokens $\mathbf{C}$ are replaced by a null embedding sequence $\emptyset$.

Because p_{anchor} and p_{cfg} are sampled independently, the network v_θ learns a joint distribution that supports four modes: unconditional generation, text-only generation, history-only forecasting, and joint text-and-history forecasting.

Inference via Clean-Timestep Anchoring. During ODE sampling, we integrate backwards from $t = 1$ to $t = 0$. Let $\hat{\mathbf{x}}_{t-\Delta t}$ be the sequence predicted by the solver at the current step. Given the user-provided clean history context $\mathbf{H}_{1:h}$ (which acts as the ground-truth $\mathbf{x}_0^{(1:h)}$), we forcefully anchor the trajectory at each solver step using the same replacement operator:

$$\mathbf{x}_{t-\Delta t} \leftarrow \mathcal{R}_{\text{rep}}(\hat{\mathbf{x}}_{t-\Delta t}, \mathbf{H}_{1:h}). \tag{S16}$$

Because the model was explicitly trained with $\tilde{\mathbf{x}}_t$ via the independent anchoring probability, this step-wise replacement introduces zero distribution shift. It guarantees that the generated future strictly adheres to the provided history context without accumulating integration errors.

Table S1: Detailed Hyperparameters of OccDirector.

Hyperparameter	Value
Model Architecture	
Hidden Dimension (d_{model})	896
Transformer Layers (Depth)	14
Attention Heads	14
Head Dimension	64
Token Refiner Layers (N_{ref})	2
Input Specifications & Tokenization	
Original Occupancy Size	$F \times 200 \times 200 \times 16$
Interpolated Occupancy Size	$F \times 224 \times 224 \times 16$
Occ-VAE Expansion Ratio (e)	8
Occ-VAE Latent Channels (C)	16
Latent Spatial Size ($H_\ell \times W_\ell$)	28×28
Patch Size (p)	2
Spatial Tokens per Frame (S)	196
Training Dynamics	
Batch Size	32
Learning Rate	1×10^{-4}
Optimizer	AdamW ($\beta_1 = 0.9, \beta_2 = 0.95$)
Training Iterations	100k
Classifier-Free Guidance (CFG) Probability	0.15
Anchoring Probability (p_{anchor})	0.5
Inference Settings	
ODE Solver	Euler
Sampling Steps (NFE)	20
Classifier-Free Guidance (CFG) Scale	7.5

Hyperparameters. We summarize the detailed hyperparameters of OccDirector in Tab. S1.

A.2 OccInteract-85k Standardization and Annotation Details

Following UniOcc [12], we adopt a unified occupancy grid of $200 \times 200 \times 16$ voxels per frame with a voxel size of 0.4 m, covering a spatial range of $[-40\,\text{m}, 40\,\text{m}]$ for the X and Y axes, and $[-3.2\,\text{m}, 3.2\,\text{m}]$ for the Z axis. Occupancy sequences are sampled at 2 Hz. For all synthetic sources, we map the raw semantic labels to the same 11 categories (including 10 semantic classes and one empty class) defined by the UniOcc protocol. The detailed mapping from source datasets, as well as several other common autonomous driving benchmarks, to our unified categories is summarized in Tab. S2.

We detail the prompt engineering process used to automatically generate the language annotations for OccInteract-85k. We employ `qwen3-vl-235b-a22b-instruct` as our frozen annotator across all data generation pipelines.

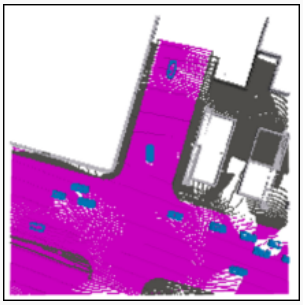
Table S2: Semantic Category Mapping for OccInteract-85k.

ID	UniOcc Category (Ours) [12]	nuScenes Category [8]	Waymo Category [8]	Carla Category [13]
0	General Object	General object, Barrier	General object, Pole, Sign	Fences, Other, Poles, Walls, Traffic signs
1	Vehicle	Bus, Car, Construction vehicle, Trailer, Truck	Vehicle	Vehicles
2	Bicycle	Bicycle	Bicycle	—
3	Motorcycle	Motorcycle	Motorcycle	—
4	Pedestrian	Pedestrian	Pedestrian, Cyclist	Pedestrians
5	Traffic Cone	Traffic cone	Traffic light, Construction cone	—
6	Vegetation	Vegetation	Vegetation, Tree trunk	Vegetation
7	Road	Drivable surface	Road	Roadlines, Roads
8	Walkable/Terrain	Sidewalk, Terrain, Other flat	Walkable	Sidewalks, Ground
9	Building	Manmade	Building	Buildings
10	Free	Free	Free	Free, Sky

Below, we present the complete prompt templates (VLM inputs) and representative output examples for the three sub-datasets:

- **OccInteract-Static:** The prompt is designed to translate 2D semantic BEV projections into ego-centric, 3D spatial layout descriptions, explicitly forbidding the mention of rendering colors.
- **OccInteract-Env:** The prompt requires the VLM to jointly reason over a front-view video sequence and a static BEV road map to deduce ego-vehicle maneuvers and describe the local topology.
- **OccInteract-Agent:** The prompt focuses on analyzing multi-agent interactions from video sequences and assigns a criticality score $(0-10)$ to enable the risk stratification discussed in the main paper.

OccInteract-Static (single-frame BEV layouts): Example Input



```
You are an AI assistant tasked with describing
     driving scenarios. Your goal is to interpret a
     2D top-down semantic projection of a 3D scene
     and generate a natural, flowing, and
     unstructured English paragraph that describes
     the underlying 3D scene's spatial layout on
     the ground plane.

This caption is for training a model to generate the
     3D occupancy grid from text, so accuracy in
     planar relationships is paramount.

**--- SCENE PROJECTION LEGEND ---**

This is the color-coding key for the top-down
     projection of the 3D scene. Use it to identify
     the objects and structures.

*   **Road:** Magenta `(255, 0, 255)`
*   **Car:** Blue `(0, 150, 245)`
*   **Pedestrian:** Red `(255, 0, 0)`
*   **Building:** Lavender `(230, 230, 250)`
*   **Vegetation:** Green `(0, 175, 0)`
*   **Terrain:** Dark Gray `(100, 100, 100)` (e.g.,
     sidewalks, curbs, dirt)
*   **Motorcycle:** Dark Orange `(200, 180, 0)`
*   **Bicycle:** Pink `(255, 192, 203)`
*   **Traffic Cone:** Light Yellow `(255, 240, 150)`
*   **Undefined (White):** Represents areas with no
     sensor data. In your description, refer to
     these as **"unseen" or "occluded" regions**.
*   **Free (Transparent/Black):** Represents
     unoccupied drivable space.


**--- CAPTIONING GUIDELINES ---**

Follow these rules precisely to generate the ideal caption:

1.  **Describe Concepts, Not Visuals:** Your task is to describe the real-world
     scene, not the rendered image.
    *   You **MUST NOT** mention colors or visual appearance details from the
     rendering.
    *   **Correct:** `The scene contains a road and a car.`
    *   **Incorrect:** `The scene has a magenta road and a blue car.`

2.  **Use Accurate Ego-Centric Directions:** All descriptions must be from the
     perspective of the ego-vehicle, located at the center of the image (indicated
     by a dark blue square) and facing forward.
    *   Image **Top** = Scene **Front / Ahead**
    *   Image **Bottom** = Scene **Rear / Behind**
    *   **Correct:** `A car is located to the rear-right.`
    *   **Incorrect:** `A car is in the bottom-right corner of the image.`

3.  **Describe Static State, Not Dynamics:** You are describing a single snapshot in
     time. **Do not infer motion, speed, or traffic flow.**
    *   Describe object placement and density, not their movement. Use terms like
     'density', 'distribution', 'placement', and 'position'.
    *   **AVOID** words like 'moving', 'stopped', 'stationary', 'traffic flow',
     'overtaking'.
    *   **Correct:** `The density of vehicles ahead is high.`
    *   **Incorrect:** `The traffic flow ahead is heavy, and the cars are
     stationary.`
```

```
4.  **Strictly Omit 3D Vertical Information:** Your description must only contain
      information about the layout on the ground (X-Y plane). **Do not** mention or
      infer object height, volume, or elevation.
    *   **Correct:** `There is a building on the right.`
    *   **Incorrect:** `There is a tall building on the right.`

5.  **Describe "Undefined" as "Unseen":** The white regions represent areas outside
      the sensor's field of view or areas occluded by other objects. Describe them
      naturally as **"unseen areas"**, **"occluded regions"**, or **"areas beyond
      sensor range"**.

---

**Output Format:**
Return the result strictly in JSON format without Markdown code block encapsulation.

```json
{
    "scene_caption": "A description of the 3D driving scene based on the BEV
      projection."
}
```

Now, generate a caption for the BEV scene projection in the image, strictly adhering
      to all guidelines and the specified multi-paragraph format.
```

OccInteract-Static: Example Output

```
{
  "scene_caption": "A scene of a multi-lane road with a complex intersection or merge
     point ahead, situated in a dense urban environment. "
}
```

OccInteract-Env (turn-point maneuver clips): Example Input




```
**Role:**
You are an expert Image Analyst specialized in autonomous driving scenarios. You
      will receive a sequence of real world (or simulation) driving video frames
      captured from the ego vehicle's front windshield, followed by a final reference
      image.
```

```
**Task:**
Analyze the ego vehicle's temporal driving behavior and the surrounding spatial
     environment based on the visual sequence and the provided road structure map.

**Input Data Description:**

1.  **Image Sequence:**
    *   The images are arranged in strictly ascending chronological order.
    *   **Frame IDs:** Displayed as black numbers in the top-left corner. Use these
     to track temporal progression.

2.  **Road Structure Map (Bird's-Eye View):**
    *   **The Final Image:** The very last image in the input list is a BEV
     (Bird's-Eye View) map representing the static road structure around the ego
     vehicle.
    *   **Temporal Alignment:** The Frame ID in the top-left of this map corresponds
     to the frame in the video sequence where this road structure applies.
    *   **Ego Vehicle Positioning:** Two perpendicular black lines indicate the ego
     vehicle's pose:
        *   **Vertical Line:** Represents the vehicle's heading (pointing Up).
        *   **Horizontal Line:** Represents the vehicle's lateral axis (Left/Right).
        *   **Intersection:** The ego vehicle is located exactly at the intersection
     of these two lines.
    *   **Semantic Colors:** The map is simplified. Only the traversable road
     surface is highlighted in **Magenta (255, 0, 255)**.

---

**Guidelines**

1.  **Behavior Classification (Taxonomy):**
    Determine the ego vehicle's action based on the sequence:
    *   **STRAIGHT:** Vehicle proceeds forward without significant deviation.
    *   **LEFT:** Vehicle executes a left turn.
    *   **RIGHT:** Vehicle executes a right turn.

2.  **Environmental Description:**
    Generate a description of the driving environment. Focus on spatial layout,
     surrounding structures (buildings), vegetation, and road morphology
     (geometry/shape).

3. **Critical Constraints (For Occupancy Labeling):**
    The output description will be used as a text label for a Occupancy  training
     pipeline. Therefore, you must adhere to the following:
    *   **Ignore Simulation Artifacts:** Even if the images appear to be from a
     simulator, describe the scene as if it were the real world. Do not mention
     "simulation," "game engine," or "synthetic data."
    *   **Focus on Geometry/Structure:** Do not include details imperceptible to a
     geometric occupancy sensor (e.g., specific weather conditions, color, texture
     details, or lighting). Focus on *what* objects are present and *where* they are.

---

**Output Format:**
Return the result strictly in JSON format without Markdown code block encapsulation.

```json
{
    "action": "STRAIGHT | LEFT | RIGHT",
    "description": "A concise summary of the ego vehicle's behavior, including the
     shape and width of the road."
}
```
```

OccInteract-Env: Example Output

```
{
    "action": "LEFT",
    "description": "The ego vehicle executes a left turn at a Y-intersection onto a
     two-lane road. The road is flanked by fields of tall crops and industrial
     structures including silos."
}
```

OccInteract-Agent (multi-agent interactions): Example Input




```
**Role:**
You are an expert Image Analyst specialized in autonomous driving scenarios. You
     will receive a sequence of real world (or simulation) driving video frames
     captured from the ego vehicle's front windshield.

**Task:**
Analyze the driving behavior, environment, and potential risks based on the provided
     frames.

**Input Data Description:**
1.  **Image Sequence:** The images are arranged in ascending chronological order.
2.  **Meta Data:** Frame IDs are displayed as Black Numbers in the top-left corner
     (use these to understand the temporal progression).

**Guidelines:**

1.  **Environment Analysis:** Analyze the situation in the sequence. Describe the
     environment, including surrounding vehicles, pedestrians, buildings,
     vegetation, road types.

2.  **Safety Scoring:** Evaluate the necessity for the ego vehicle (driver) to
     exercise caution or take emergency action. Assign a **score (0-10)** based on
     the following scale:
    *   **0-2:** Safe driving conditions. Good visibility, no immediate hazards
     (e.g., clear road).
    *   **3-5:** Caution required. Vulnerable road users (pedestrians, cyclists) or
     potential obstacles present in the field of view.
    *   **6-8:** Immediate action/Emergency required. High-risk situations (e.g.,
     vehicle cutting in, sudden braking ahead, jaywalking).
    *   **9-10:** Collision inevitable or has occurred.

3.  **Critical Constraints (For Occupancy Labeling):**
    The output description will be used as a text label for a Occupancy  training
     pipeline. Therefore, you must adhere to the following:
```

```
    *   **Ignore Simulation Artifacts:** Even if the images appear to be from a
      simulator, describe the scene as if it were the real world. Do not mention
      "simulation," "game engine," or "synthetic data."
    *   **Focus on Geometry/Structure:** Do not include details imperceptible to a
      geometric occupancy sensor (e.g., specific weather conditions, color, texture
      details, or lighting). Focus on *what* objects are present and *where* they are.

**Output Format:**
Return the result strictly in JSON format. Do not output markdown code blocks alone,
     just the raw JSON string.

```json
{
    "description": "Briefly describe the ego vehicle's driving behavior and the
     cause of risks (if any).",
    "score": <Integer between 0-10>
}
```
```

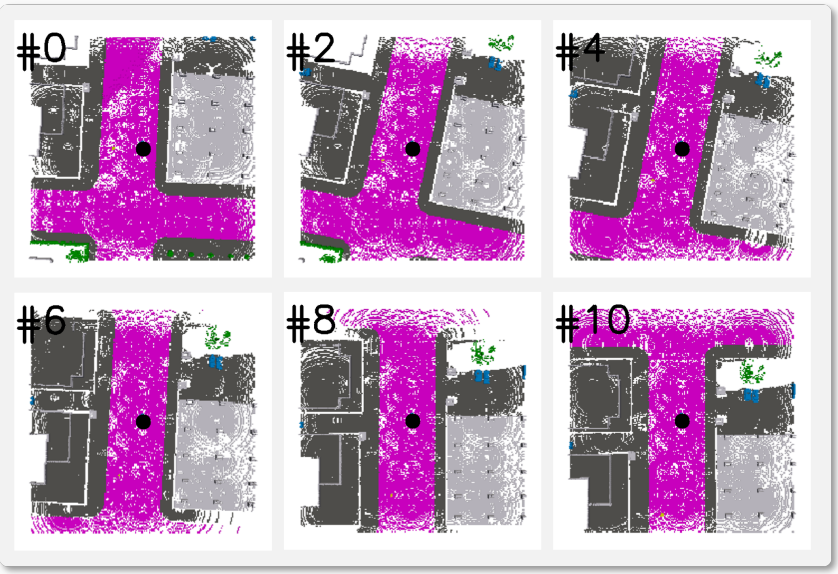
OccInteract-Agent: Example Output

```
{
  "description_short": "The ego vehicle follows an ambulance which suddenly spins out
     and blocks the road, forcing the ego vehicle to take emergency evasive action.",
  "score": 8
}
```

A.3 Prompt Templates for Automated Quality Assessment

We provide the complete rubric-based judge prompt used in "Automated Quality Assessment". The prompt is applied to a BEV image sequence rendered from the semantic occupancy grid and a text description (the caption or generation prompt). Below, we present the system message defining the evaluation rubric and a representative output example generated by the VLM judge.

Automated Quality Assessment: Example Input



Related prompt: The scene is an urban environment with multi-lane roads and tall, multi-story buildings lining both sides. The ego vehicle is positioned in the center lane and proceeds straight ahead. No other vehicles, pedestrians, or moving objects are visible in the field of view at any point in the sequence. The road markings are clear, and the path is unobstructed. The surrounding infrastructure includes streetlights, a low wall on the right, and some small structures on the sidewalks. The overall situation is static and poses no immediate threat.

```
You are an expert evaluator for an autonomous driving scene generation system. Your
     task is to score generated Bird's-Eye View (BEV) driving video frames based on
     three criteria: Scene Completeness, Structural Correctness, and Instruction
     Following.

**Input Data Description:**

(1) A text Prompt that was used to generate the scene.

(2) Bird's-Eye View Image Sequence (video frames):
    - Images are in strictly ascending chronological order.
    - Frame IDs are shown as black numbers in the top-left corner for tracking
     temporal progression.

**Scene projection legend (BEV semantic colors):**
- Road: Magenta (255, 0, 255)
- Car: Blue (0, 150, 245)
- Pedestrian: Red (255, 0, 0)
- Building: Lavender (230, 230, 250)
- Vegetation: Green (0, 175, 0)
- Terrain: Dark Gray (100, 100, 100)
- Undefined (White): unseen/empty areas where the model generated nothing.
- Free (Transparent/Black): unoccupied drivable space.

**Ego vehicle positioning:** a black point at the center indicates the ego pose; the
     area above it is the front of the vehicle.

**Evaluation criteria & scoring rubrics (1-5 points):**
```

```
1) Scene Completeness. Assess how much of the scene is filled versus being empty
     (white/undefined), ignoring image edges.
    - 1 (Bad): almost entirely white/empty; failed to generate meaningful content.
    - 2 (Poor): large unnatural white patches dominate the scene.
    - 3 (Fair): noticeable empty areas that hurt realism.
    - 4 (Good): mostly complete; white areas are minimal and acceptable.
    - 5 (Excellent): fully rendered without unnatural empty patches.

2) Structural Correctness. Assess physical plausibility and topological consistency
     independent of the prompt.
    - 1 (Bad): nonsensical layout (e.g., buildings on roads, chaotic placement).
    - 2 (Poor): major flaws (illogical breaks, strange relationships).
    - 3 (Fair): plausible but contains minor unrealistic elements.
    - 4 (Good): logical and consistent driving environment.
    - 5 (Excellent): highly realistic and coherent complex layout.

3) Instruction Following. (Most important) Assess how well the BEV sequence matches
     the given prompt.
    - 1 (Bad): ignores/contradicts the prompt.
    - 2 (Poor): captures only a minor element; misses the main theme.
    - 3 (Fair): matches general theme but misses key details.
    - 4 (Good): generates the described scene and primary objects; may miss minor
     nuances.
    - 5 (Excellent): perfect match, including specified elements and spatial
     relations.

**Additional instructions:** ignore any color information mentioned in the prompt
     (e.g., traffic light state or vehicle paint color), since the BEV semantic
     rendering uses a fixed color legend.

**Output format:** You must respond with a single JSON object and no surrounding
     text.

{
  "thought": "Brief rationale for the scores.",
  "scores": {
    "completeness_score": 1-5,
    "structural_correctness_score": 1-5,
    "instruction_following_score": 1-5
  }
}

Now, evaluate the provided BEV image sequence based on the prompt.
```

Automated Quality Assessment: Example Output

```
{
  "thought": "1. Scene Completeness: The generated scene is well-populated. The road,
     terrain, and building areas are clearly defined across all frames. While there is
     some minor 'salt-and-pepper' noise within the semantic regions (common in
     occupancy grid projections), there are no large, unnatural empty patches that
     disrupt the scene's realism. Score: 4.
  2. Structural Correctness: The scene exhibits excellent temporal and spatial
     consistency. As the ego vehicle moves forward (indicated by the frame sequence),
     the environment scrolls backward smoothly and logically. The road geometry,
     transitioning from a straight segment to a T-intersection, is physically plausible
     and represents a realistic urban layout. Score: 5.
  3. Instruction Following: The image follows the core instruction of generating an
     urban environment where the ego vehicle proceeds straight ahead on an unobstructed
     path. However, it fails on specific constraints. The prompt explicitly states,
     \"No other vehicles, pedestrians, or moving objects are visible,\" yet there are
     distinct blue blocks (cars) visible on the right side (likely parked) and a
```

```
    persistent red speck (pedestrian) in the road area in multiple frames.
    Additionally, the prompt describes buildings \"lining both sides,\" but the
    semantic segmentation shows buildings (lavender) primarily on the right, while the
    left side is dominated by terrain (dark gray). Score: 3.",
  "scores": {
    "completeness_score": 4,
    "structural_correctness_score": 5,
    "instruction_following_score": 3
  }
}
```


A.4 Evaluation Protocol and Metric Definitions

Rendering occupancy to BEV RGB. Following previous work [2,6,15], we render the 3D/4D occupancy grids into BEV-view RGB images and clips. These rendered RGB representations are subsequently utilized to evaluate all single-frame and video-level visual quality metrics.

Data preparation for visual metrics. To evaluate the visual quality and distributional realism of the generated samples, we construct a standardized evaluation set. Specifically, we uniformly sample 10K ground-truth 16-frame occupancy clips from the dataset. For each evaluated method, we generate 10K corresponding 16-frame clips using the classifier-free guidance (CFG) unconditional branch. For single-frame metrics, we uniformly sample 5 frames from each clip, whereas video-level metrics are computed using the full 16-frame sequences.

Single-frame visual quality. Using the sampled 5 frames for each clip, we compute Inception Score (IS) [1], Fréchet Inception Distance (FID) [5], and Kernel Inception Distance (KID) [3] to assess per-frame fidelity. We also report image-level Precision and Recall computed via pre-trained Inception-v3 [7] features to explicitly disentangle sample fidelity and distributional diversity.

Video realism and diversity. To evaluate temporal coherence and dynamic realism, we compute the Fréchet Video Distance (FVD) [9] between the generated and ground-truth 16-frame video distributions. We also report video-level Precision and Recall computed using I3D [4] features to measure spatio-temporal fidelity and diversity.

VLM-based instruction alignment score. We adopt the rubric-based VLM evaluator in "Automated Quality Assessment" to quantify instruction alignment of generated sequences. The evaluator assigns integer scores in $[1, 5]$ (higher is better) along three axes: scene completeness, structural correctness, and instruction following, and we report their mean. To ensure consistent and rigorous comparison, we fix `gemini-3.0-pro` as our sole judge VLM for all experiments. We select this model specifically for its high sensitivity to the *Instr. follow* axis, which is the primary focus of our text-guided generation task, ensuring that the scores faithfully reflect the alignment between prompts and generated 4D occupancies.

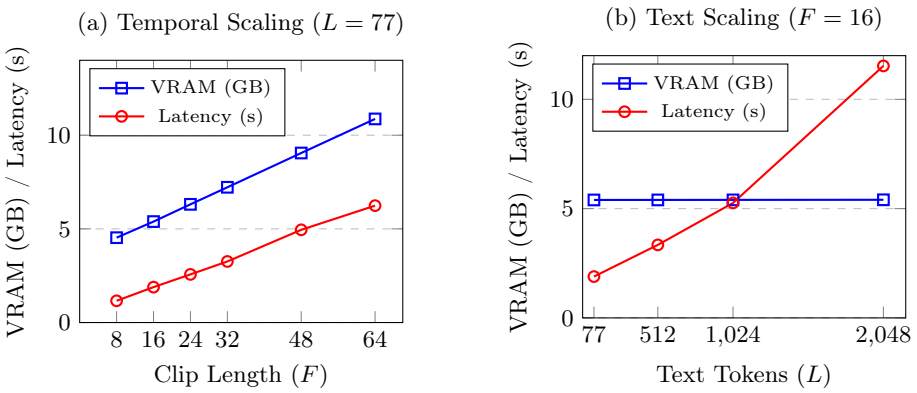


Fig. S1: Inference resource scaling. (a) VRAM and latency scale linearly with frames F. (b) VRAM is invariant to text length L as the memory peak is dominated by the occupancy stream's temporal attention. Measurements on NVIDIA A100 (80GB) using 20 Euler steps.

B Additional Quantitative Results

B.1 Inference Resource Consumption

We evaluate the inference efficiency of OccDirector across varying output temporal lengths (F) and input text token counts (L). As shown in Fig. S1, our Spatio-Temporal Separated Attention (STSA) maintains a strictly linear memory growth relative to clip length. Notably, the peak VRAM remains remarkably invariant to the text token length L during 4D generation ($F \geq 16$).

This memory robustness stems from our decoupled attention design: the memory-intensive temporal attention blocks exclusively process the occupancy stream, while the cross-modal spatial interactions (where L is involved) consume significantly less memory than the temporal bottleneck. Consequently, increasing L from 77 to 2048 yields a negligible VRAM increase ($< 0.1\%$), enabling OccDirector to handle highly complex behavioral instructions without additional memory overhead.

B.2 Architectural Sensitivity

We investigate the impact of backbone depth, refiner complexity, and text encoder selection on model performance.

Backbone and Refiner Depth. As shown in Table S3, increasing MMDiT layers from 10 to 14 yields consistent gains in FVD, confirming that larger model capacity is beneficial for capturing complex 4D dynamics. For the Token Refiner, we observe that a depth of 2 layers is sufficient; further increasing it to 4 layers yields diminishing returns, suggesting that the bottleneck lies in the generative backbone rather than the alignment bridge.

Table S3: Architectural Sensitivity. We report FVD and Mean VLM score for various model depths and text encoders.

Variant	MMDiT Layers	Refiner Depth	FVD ↓	VLM Mean ↑
Small	10	1	512	2.12
Base	12	2	339	3.18
Large (Ref1)	14	1	298	3.42
Large (Ours)	14	2	275	**3.73**
Large (Ref4)	14	4	**274**	3.72

Table S4: Comparison of different VLM encoders. Performance is evaluated on the full OccInteract-85k validation set.

VLM Encoder Type	FVD ↓	Semantic Align. ↑	Mean VLM Score ↑
Qwen2.5-VL-7B	282	**3.33**	3.69
Qwen2.5-VL-3B	315	3.02	3.35
Qwen3-VL-2B (Ours)	**275**	3.32	**3.73**

VLM Type. We compare different VLM encoders in Table S4. Our default Qwen3-VL-2B refiner achieves the best FVD and Mean VLM Score, even outperforming the larger Qwen2.5-VL-7B. This suggests that the improved visual-linguistic grounding capabilities of the latest VLM generation are more critical for 4D occupancy alignment than raw parameter count.

B.3 Methodological Ablations

History-Prefix Anchoring Ratio. The anchoring ratio γ controls the guidance strength from the past during the flow-matching process. Table S5 shows that $\gamma = 0.50$ provides the optimal balance. A lower ratio leads to temporal drifting (higher FVD), while an excessively high ratio ($\gamma = 0.75$) slightly degrades both visual quality and semantic alignment as the model becomes over-constrained by the history prefix.

Sensitivity to CFG Scale across Sub-datasets. We analyze the instruction alignment (VLM Mean) across three sub-corpora under different Classifier-Free Guidance (CFG) scales. As shown in Table S6, different tasks exhibit varying sensitivities to CFG. Dynamic scenes (OccInteract-Env and OccInteract-Agent) benefit from a higher CFG scale ($s = 7.5$) to ensure precise spatio-temporal coordination, whereas OccInteract-Static performance peaks earlier at $s = 5.0$ and declines as the guidance strength further increases.

B.4 Robustness of VLM-based Evaluation

To further validate the objectivity of the instruction alignment scores, we extend the evaluation conducted in the main experiments to include two addi-

Table S5: Ablation on History-Prefix Anchoring Ratio γ.

Ratio γ	0.0 (None)	0.25	0.50 (Ours)	0.75
FVD ↓	354	311	**275**	288
VLM Mean ↑	3.43	3.55	**3.73**	3.68

Table S6: CFG scale (s) sensitivity per sub-dataset. We report the VLM Mean score.

Sub-dataset	$s = 1.0$ (None)	$s = 5.0$	$s = 7.5$ (Ours)	$s = 10.0$
OccInteract-Static	2.52	**3.72**	3.55	3.34
OccInteract-Env	2.24	3.45	**4.02**	3.51
OccInteract-Agent	1.85	3.10	**3.62**	3.25
Mean ↑	2.20	3.42	**3.73**	3.37

tional state-of-the-art VLMs: `gpt-5.2` and `qwen3-vl-235b-a22b-instruct`. As discussed in the dataset validation section, while different models exhibit systematic scoring offsets, the relative performance trend across methods remains highly consistent.

As shown in Table S7, OccDirector consistently outperforms all text-adapted baselines across all three judges. Notably, even under the strictest evaluator (`gpt-5.2`), our full model maintains a significant margin over the strongest baseline (2.82 vs. 2.18), confirming that the improvements in instruction alignment are not model-specific but represent a general advancement in text-to-4D occupancy synthesis.

B.5 User Study

Following OccSora [11] and DynamicCity [2], we conduct a user study to validate our VLM-based automated evaluation. The study includes 30 samples in total, with 10 samples randomly drawn from each method. We recruited 10 human participants to rate the clips. Each participant was asked to rate the generation clips on a 1–5 Likert scale following the same rubric as our VLM scorer (*i.e.*, scene completeness, structural correctness, and semantic alignment).

The results in Table S8 confirm that OccDirector is strongly preferred by human evaluators. Most importantly, the high Pearson correlation coefficient ($\rho = 0.88$) between human ratings and our automated VLM scores justifies the reliability of the latter as a primary evaluation metric for large-scale 4D occupancy generation.

C Additional Qualitative Results

In this section, we provide a deeper interpretation of the qualitative results, demonstrating how OccDirector effectively bridges the semantic-spatiotemporal

Table S7: Cross-VLM Evaluation Robustness. We report the Mean VLM score across different judge models. Despite the variance in absolute values, the performance gain of OccDirector remains consistent across all evaluators.

Judge VLM	DynamicCity + CLIP	DynamicCity + T5	OccDirector (w/o HP)	OccDirector (Full)
`gemini-3.0-pro` (Main)	3.09	2.97	3.43	**3.73**
`gpt-5.2` (Strict)	2.18	2.11	2.55	**2.82**
`qwen3-vl-235b-a22b-instruct`	2.58	2.44	2.92	**3.24**

Table S8: User study results and Correlation. Human scores are averaged across participants. ρ denotes the Pearson correlation coefficient between human and VLM scores.

Metric	DynamicCity + CLIP	OccDirector (w/o HP)	OccDirector (Full)	Correlation ρ
Completeness	2.85	3.45	**3.78**	0.88
Structural	3.24	3.62	**3.95**	0.85
Sem. Align.	2.55	3.10	**3.82**	0.91
Mean	2.88	3.39	**3.85**	**0.88**

gap. In all following figures, the ego vehicle is marked with a *black car icon*. Specific visual highlights, such as newly appearing hazards, pedestrians, or intended trajectories, are indicated with *red circles* or *red dashed lines*.

C.1 Dataset Samples and Criticality Analysis

We visualize representative samples from the OccInteract-Agent corpus in Fig. S2, categorized by their VLM-predicted criticality scores $s \in [0, 10]$. The samples demonstrate the dataset's ability to cover the full safety-critical spectrum, which is essential for providing diverse supervision during training.

- **Low Criticality (Score 1):** These samples represent routine driving on urban roads. The dataset captures stable 4D occupancy volumes for the ego vehicle and surrounding static elements, providing high-fidelity geometric priors for temporally consistent and physically plausible scenes in low-conflict environments.
- **Medium Criticality (Score 4):** In these scenarios, the corpus contains potential conflicts that require behavioral adaptation. For example, when a lead van executes a left turn across the ego's path, the captured occupancy sequence and corresponding caption accurately reflect the interactive dynamics and the necessary spatial buffers.
- **High Criticality (Score 7):** These cases highlight imminent hazards within the training data. In the crash scenario, the dataset accurately captures the

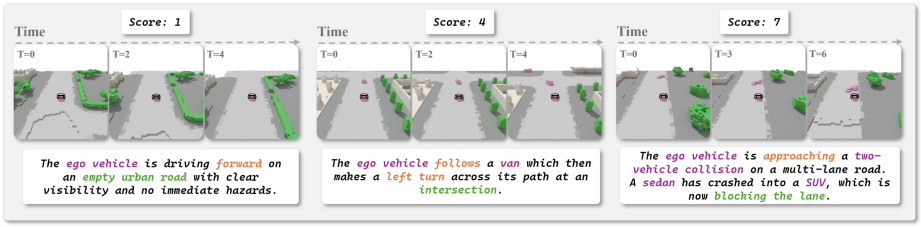


Fig. S2: Qualitative samples from OccInteract-Agent across criticality score regimes. The samples progress from routine driving (left) to potential conflicts (middle) and imminent hazards (right), demonstrating the breadth of safety-critical scenarios covered by our automated data engine.

irregular and complex occupancy volumes of the colliding vehicles and the subsequent lane blockage. Such samples are crucial for teaching the model to represent and reason about rare, safety-critical edge cases.

C.2 Hierarchical Generation

Fig. S3 showcases the precision and diversity of OccDirector across three levels of hierarchical control, highlighting its superior semantic-to-spatial mapping.

(a) **Static Environment Description:** The model translates abstract descriptors (e.g., "natural environment" vs. "busy urban intersection") into geometrically distinct road topologies. Note how the model preserves fine-grained road boundaries and parked vehicle distributions, maintaining high-fidelity layout priors independent of temporal dynamics.
(b) **Agent-Environment Interaction:** OccDirector generates affordance-aware ego-trajectories that are strictly grounded in the generated layout. Whether performing a right turn at a crossroads or proceeding through a T-intersection, the ego's 4D volume remains logically aligned with the drivable road regions.
(c) **Multi-Agent Interaction:** The model orchestrates complex causal behaviors between the ego and other agents. In the emergency braking scenario, the model synthesizes a sudden obstacle and the corresponding rapid deceleration of the ego car, demonstrating an integrated understanding of dynamic agent logic and spatial constraints.

C.3 History-Prefix Control

To evaluate temporal consistency and causal logic over long horizons, we analyze multi-stage narratives in Fig. S4. These samples highlight how our history-prefix anchoring prevents "scene-drifting" and ensures logical transitions.

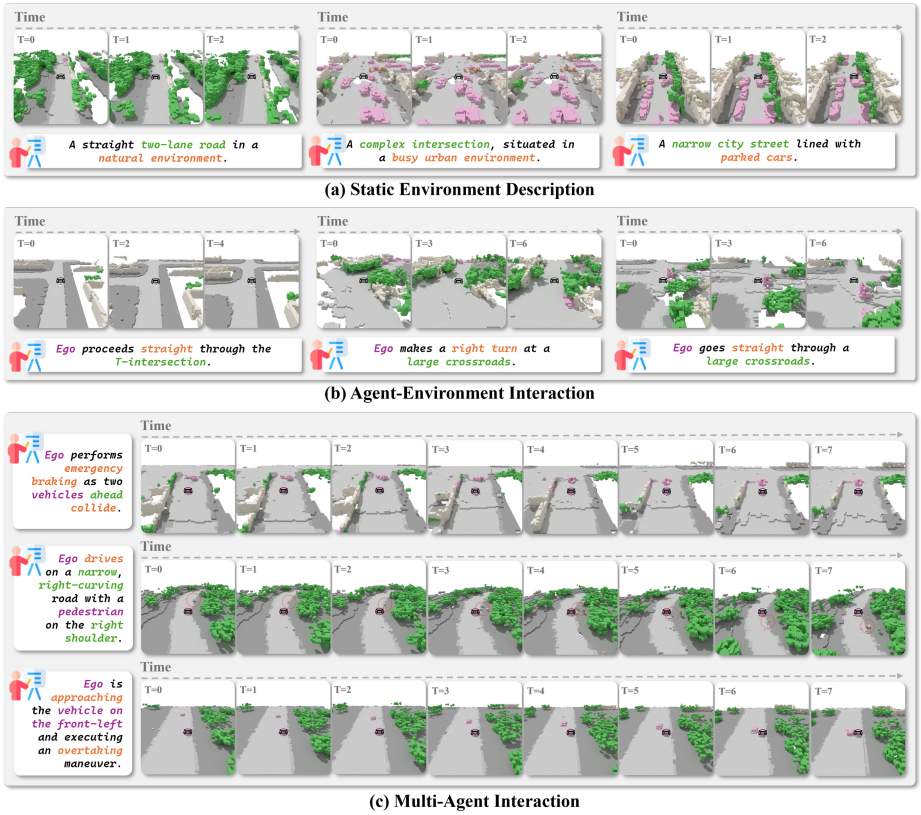


Fig. S3: Additional results on hierarchical generation control. We visualize (a) static layout synthesis, (b) behavior-conditioned ego dynamics, and (c) multi-agent reactive orchestration.

- **Handling Occlusions and Transitions (Top):** The model demonstrates sophisticated spatial reasoning. As the ego overtakes a bus in Stage 1, a previously occluded car is revealed (marked with a *red circle*). In Stage 2, the model smoothly transitions this revealed car into a lateral cut-in hazard. This confirms that OccDirector maintains a persistent internal representation of the scene, even for objects that are partially hidden or suddenly visible.
- **Persistent Scene Context (Middle):** During the decisive overtaking maneuver, the model maintains perfect spatial continuity of the bus and the background road layout across stages. The new instruction ("decisively overtake") is seamlessly integrated with the historical context without altering the pre-established environment.
- **Grounding Procedural Logic (Bottom):** This complex stop-and-go sequence (waiting for a light → proceeding → stopping behind a bus) illustrates the model's grasp of procedural semantics. Unlike standard text-to-video

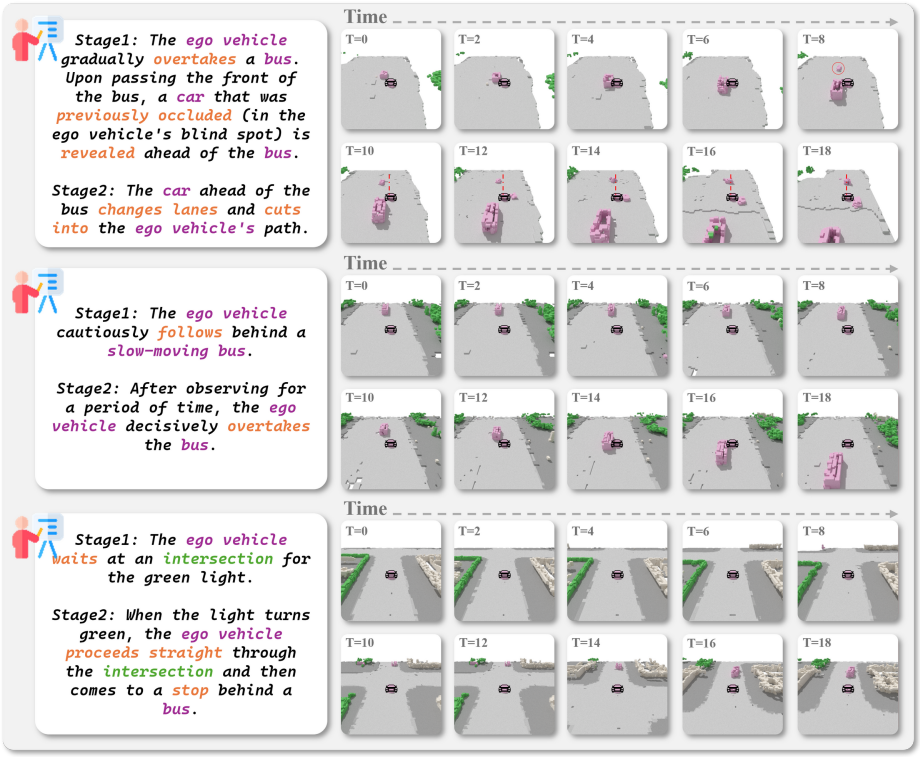


Fig. S4: Long-horizon narrative generation with history-prefix anchoring. These multi-stage rollouts demonstrate the model's ability to maintain spatio-temporal coherence and respond to evolving instructions without losing scene consistency.

models that often struggle with temporal ordering, OccDirector executes these logically dependent stages while maintaining a single, consistent 4D occupancy volume.

C.4 Fine-grained Text-to-Occupancy Alignment

We provide additional qualitative comparisons in Fig. S5 to further demonstrate OccDirector's superior text-to-occupancy alignment compared to the DynamicCity + CLIP baseline.

(a) **Negation and Scene Density Control:** We evaluate the model's ability to process negative constraints (e.g., "no other vehicles") and control scene density. While OccDirector faithfully renders the requested environment, the baseline (DynamicCity + CLIP) fails to respect the negation, consistently hallucinating busy traffic in an "empty intersection".

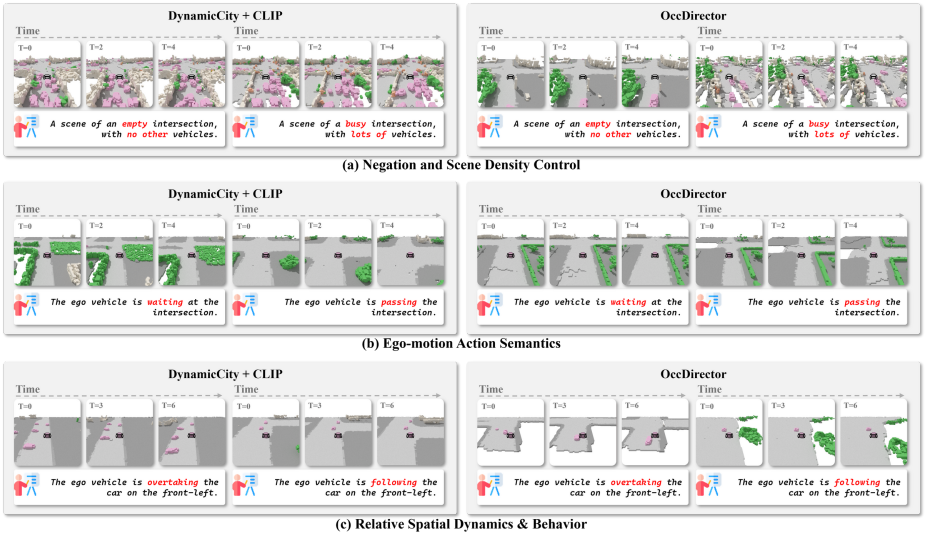


Fig. S5: Additional qualitative comparisons on fine-grained text alignment. We showcase OccDirector's ability to handle (a) negation and scene density, (b) ego-motion action semantics, and (c) complex relative spatial dynamics between agents.

(b) **Ego-motion Action Semantics:** The model correctly distinguishes between the ego vehicle "waiting" versus "passing" an intersection. In contrast, the baseline exhibits attribute confusion, depicting a moving ego vehicle even when the prompt explicitly requests it to wait, failing to decouple these distinct temporal states.

(c) **Relative Spatial Dynamics and Behavior:** OccDirector captures subtle interaction logic between agents. In the "overtaking" case, it synthesizes a precise lateral maneuver relative to a moving agent, whereas the baseline erroneously overtakes a static parked vehicle. For "following", OccDirector maintains stable longitudinal positioning, while the baseline fails to coordinate the relative spatial dynamics, generating extraneous vehicles that disrupt the intended interaction (*i.e.*, additional car directly on the front).

D Potential Societal Impact & Limitations

D.1 Potential Societal Impact

Our work, OccDirector, contributes to the development of safer and more reliable autonomous driving (AD) systems by providing a high-fidelity, controllable generative model for 4D occupancy. By enabling the synthesis of diverse, safety-critical traffic scenarios from simple natural language prompts, it allows AD systems to be rigorously tested and validated against rare, high-risk edge cases

that are difficult and dangerous to capture in the real world. This democratizes access to high-quality 4D training data for research institutions that may not have the resources for large-scale data collection.

However, like any generative technology, OccDirector could be misused. The ability to create realistic 4D traffic scenarios might be exploited to generate deceptive or "deepfake" data, potentially misleading safety evaluations or creating unrealistic performance benchmarks. Furthermore, an over-reliance on synthetic data could lead to models that fail to generalize to the full complexity of real-world environments, particularly if the generated distribution has subtle biases. We advocate for the responsible use of synthetic data as a supplement to, rather than a replacement for, real-world validation.

D.2 Limitations

Despite the advancements in OccDirector, several limitations remain:

- **Data Artifacts:** Since OccDirector is trained on real-world logs (e.g., nuScenes, Waymo), it inherits inherent sensor artifacts, such as blind spots and missing ego-vehicle voxels, which may be reflected in the generated occupancy grids.
- **Long-horizon Consistency:** While history-prefix anchoring significantly improves temporal stability, generating extremely long sequences (beyond 8s) still faces challenges in maintaining absolute physical consistency due to the accumulation of small errors over time.
- **VLM-Judge Bias:** Our evaluation relies on rubric-based VLM judges (e.g., GPT-5.2, Gemini-3.0-pro). These models may exhibit subjective biases or differing calibration scales (e.g., GPT-5.2 being noticeably stricter than Gemini-3.0-pro), which can affect the absolute scores in human-like semantic assessment.
- **Semantic Granularity:** Our current model operates on 11 semantic categories. It does not yet capture fine-grained object states, such as vehicle turn signals, brake lights, or subtle pedestrian gestures, which are crucial for high-level interaction understanding.

E Public Resources Used

In this section, we acknowledge the public datasets, models, and open-source libraries used in this research.

E.1 Data Source & Privacy Declaration

Declaration on Data Usage and Privacy: All data utilized in this study are derived strictly from publicly available open-source benchmarks and purely synthetic/simulation environments. No proprietary commercial data, vehicle onboard telemetry, or private real-world driving data from Li Auto or any other entity were used or accessed in this research.

- UniOcc[4] ..... MIT License

[4] https://github.com/tasl-lab/uniocc

E.2 Public Models & Benchmarking Vision-Language Models

Declaration on Model Usage: The third-party proprietary and open-source Vision-Language Models (VLMs) and foundation models listed below are employed solely as public academic benchmarking baselines and data annotation tools.

- Qwen3-VL[5] ..... Apache 2.0
- OpenAI CLIP[6] ..... MIT License
- OpenAI GPT-5.2[7] ..... Proprietary
- Google T5[8] ..... Apache 2.0
- Google Gemini-3.0-pro[9] ..... Proprietary

E.3 Public Implementations & Libraries

Declaration on Code and Software Usage: The open-source implementations, simulation tools, and third-party libraries listed below were utilized strictly for academic experimentation, validation, and paper implementation within this research project. None of these external codebases or dependencies are integrated into or affiliated with Li Auto's commercial software stack or production vehicle systems.

- CarlaSC[10] ..... MIT License
- CARLA Simulator[11] ..... MIT License
- TTSG[12] ..... No explicit license
- PyTorch Lightning[13] ..... Apache 2.0
- DOME[14] ..... Apache 2.0
- COME[15] ..... Apache 2.0
- DynamicCity[16] ..... No explicit license

References

1. Barratt, S., Sharma, R.: A note on the inception score. arXiv preprint arXiv:1801.01973 (2018)

[5] https://github.com/QwenLM/Qwen3-VL
[6] https://github.com/openai/CLIP
[7] https://openai.com
[8] https://github.com/google-research/text-to-text-transfer-transformer
[9] https://deepmind.google/technologies/gemini
[10] https://github.com/UMich-CURLY/3DMapping
[11] https://github.com/carla-simulator/carla
[12] https://github.com/basiclab/TTSG
[13] https://github.com/Lightning-AI/pytorch-lightning
[14] https://github.com/gusongen/DOME
[15] https://github.com/synsin0/COME
[16] https://github.com/3DTopia/DynamicCity

2. Bian, H., Kong, L., Xie, H., Pan, L., Qiao, Y., Liu, Z.: Dynamiccity: Large-scale 4d occupancy generation from dynamic scenes. arXiv preprint arXiv:2410.18084 (2024)
3. Bińkowski, M., Sutherland, D.J., Arbel, M., Gretton, A.: Demystifying mmd gans. arXiv preprint arXiv:1801.01401 (2018)
4. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
5. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems **30** (2017)
6. Lee, J., Lee, S., Jo, C., Im, W., Seon, J., Yoon, S.E.: Semcity: Semantic scene generation with triplane diffusion. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 28337–28347 (2024)
7. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016)
8. Tian, X., Jiang, T., Yun, L., Mao, Y., Yang, H., Wang, Y., Wang, Y., Zhao, H.: Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. Advances in Neural Information Processing Systems **36**, 64318–64330 (2023)
9. Unterthiner, T., Van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., Gelly, S.: Towards accurate generative models of video: A new metric & challenges. arXiv preprint arXiv:1812.01717 (2018)
10. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
11. Wang, L., Zheng, W., Ren, Y., Jiang, H., Cui, Z., Yu, H., Lu, J.: Occsora: 4d occupancy generation models as world simulators for autonomous driving. arXiv preprint arXiv:2405.20337 (2024)
12. Wang, Y., Huang, X., Sun, X., Yan, M., Xing, S., Tu, Z., Li, J.: Uniocc: A unified benchmark for occupancy forecasting and prediction in autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 25560–25570 (2025)
13. Wilson, J., Song, J., Fu, Y., Zhang, A., Capodieci, A., Jayakumar, P., Barton, K., Ghaffari, M.: Motionsc: Data set and network for real-time semantic mapping in dynamic environments. IEEE Robotics and Automation Letters **7**(3), 8439–8446 (2022)
14. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al.: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025)
15. Yang, Y., Liang, A., Mei, J., Ma, Y., Liu, Y., Lee, G.H.: X-scene: Large-scale driving scene generation with high fidelity and flexible controllability. arXiv preprint arXiv:2506.13558 (2025)